

An Adaptive & Pre-Emptive Domain Name System (DNS) Enhancement System

Field of the Invention

This invention relates to a system and a method of performing pre-emptive
5 operations. Particularly, this invention relates to a system and a method of determining possible data required and fetching of data in advance.

Prior Art

Figure 1 illustrates an example of a communication system comprising a client
10 110, a router 120, a network 130, DNS server 1 140, and DNS server 2 150. The client 110 is connected to the network 130 via a router 120. In a typical network traffic flow, a user initiates a connection to a website by entering a domain name in the website address in a browser. With the domain name, the client 110 sends a query to DNS server 1 140 over the network 130. Assuming that the domain name is not stored on DNS server
15 1 140, the DNS server 1 140 would forward the query to another DNS server, in this case, DNS server 2 150. Assuming DNS server 2 150 has the IP address of the domain name, DNS server 2 150 would forward the IP address of the domain name to DNS server 1 140 and in turn DNS server 1 140 would forward the IP address of the domain name to Client 110. Upon receipt of the IP address, the client would perform a TCP 3-way handshake to
20 establish connection with the server of the website. Subsequently, the client would send HTTP request to the server to obtain the content of the website. In turn, the server sends HTTP reply to client to display website and client continues to send more DNS query to DNS server for IP addresses before the client can fetch the remaining content from the server. As you may note, to display all the content of a webpage, a client typically sends
25 multiple DNS queries to the DNS server to obtain content for one website. This is inefficient and unnecessarily consumes resources.

Hence, those skilled in the art are striving to provide an improved system and method to improve the DNS performance.

5

Summary of the Invention

The above and other problems are solved and an advance in the art is made by a system for performing pre-emptive operations in accordance with this invention. A first advantage of the system for performing pre-emptive operations in accordance with this invention is that the system for performing pre-emptive operations improves enterprise
10 network performance. A second advantage of the system for performing pre-emptive operations in accordance with this invention is that the system for performing pre-emptive operations optimises DNS traffic performance. Other advantages can be gleaned from the detailed description below.

15

In accordance with an aspect of the invention, a system for performing pre-emptive operations is provided in the following manner. The system comprises an enhancer installed on a client operating system. The enhancer comprises a monitoring engine communicatively connected to TCP/IP stack and processes, a database and decision engine having a database storing a profile data structure and an event data
20 structure, a controller executing instructions received from the database and decision engine, and instructions stored on the database. The instructions stored on the database comprises the following steps: determining a profile of a user based on the profile data structure, retrieving an event list from the event data structure associated to the determined profile, monitoring an event, computing a probability score for the event,
25 determining whether the probability score is above a predetermined threshold; and executing pre-emptive operations in response to the probability score being above the predetermined threshold.

In accordance with an embodiment of the invention, the instruction to determining the profile of a user based on the profile data structure comprises retrieving a parameter stored on a client operating system, matching the parameter with parameters in the profile data structure, and determining a profile in the profile data structure having a parameter that matches the retrieved parameter.

In accordance with an embodiment of the invention, the event list in the event data structure is divided into two groups, namely, a first group being an active group and a second group being a reserve group. In accordance with an embodiment of this embodiment, the instruction to monitoring an event further comprises clearing a probability score stored on the database, detecting an event, matching the event to an action in the active group, incrementing a counter of the matched action, and computing the probability score for the event. Further, the instruction to computing the probability score for the event comprises retrieving a weightage of the matched action, and adding the weightage to the probability score.

In accordance with an embodiment of the invention, the instruction to monitoring an event further comprises matching the event to one of the actions in the reserve group in response to the event not being able to match with any one of actions in the active group, and adding a new entry in the reserve group in response to the event not being able to match with any one of actions in the reserve group. The instruction to monitoring an event further comprises incrementing a counter of the matched action in the reserve group in response to the event matches the action in the reserve group, determining whether the counter of the matched of action in the reserve group is more than any one of the counter of the actions in the active group, and updating the actions in the active group

in response to the counter of matched action in the reserve group being more than one of counter of the action in the active group.

In accordance with an embodiment of the invention, the instruction to updating the actions in the active group further comprises replacing an action in the active group with the matched action in the reserve group having a counter that is more than the counter of the action in the active group, and apportioning a weightage among the actions in the active group. The weightage may be apportioned according to the value of the counters of each action in the active group.

In accordance with an embodiment of the invention, the system comprises a second enhancer installed on a server communicatively connected to the client operating system. The second enhancer comprises a monitoring engine communicatively connected to TCP/IP stack of the server, a database and decision engine having a database storing a profile data structure and an event data structure, a controller executing instructions received from database and decision engine, and instructions stored on the database to perform the following steps determining a profile of a user based on the profile data structure, retrieving an event list from the event data structure associated to said determined profile, monitoring an event, computing a probability score for the event, transmitting the probability score to the enhancer installed on the client operating system.

In accordance with an embodiment of the invention, the instruction to computing a probability score for the event stored on the enhancer installed on the client operating system comprises retrieving a weightage for the probability score, applying the weightage of the probability score to the probability score to determine a first score, retrieving a weightage of the probability score of the server; applying the weightage of the probability

score of the server to the probability score received from the second enhancer to determine a second score, adding the first and second scores to determine a net probability score, and executing the pre-emptive operations in response to the net probability score being above the predetermined threshold.

5

A method in accordance with the system described is also provided.

Brief Description of the Drawings

The above and other features and advantages in accordance with this invention
10 are described in the following detailed description and are shown in the following drawings:

Figure 1 illustrating an example of a communication system;

Figure 2 illustrating a software architecture of a system that executes the processes in accordance with an embodiment of this invention;

15 Figure 3 illustrating a flow diagram of a process performed by the system being installed on an operating system or a router/server in accordance with an embodiment of this invention;

Figure 4 illustrating a flow diagram of a process performed by the system being installed on a operating system in accordance with an embodiment of this invention; and

20 Figure 5 illustrating a timing diagram of a process performed by the system being installed on both the operating system and router/server in accordance with an embodiment of this invention.

Detailed Description

25 This invention relates to a system and a method of performing pre-emptive operations. Particularly, this invention relates to a system and a method of determining possible data required and fetching of data in advance.

Domain Name System (DNS) is a naming system which assigns domain names to each of the entities. More particularly, DNS translates domain names which can be easily recognised by humans, to the numerical IP addresses.

5

For purpose of this discussion, content may be executable files such as object code, scripts, and patches or non-executable files such as documents that may need to be opened or other data files that need to be accessed.

10

Figure 2 illustrates the block diagram of the system 200 in accordance with an embodiment of this invention. The system 200 comprises an enhancer module 205 having a monitoring engine 210, a database and decision engine 220, and a controller 230.

15

The monitoring engine 210 is a module that monitors various system statuses such as running processes, network status, IP addresses, Wifi SSID, DHCP status etc. The monitoring engine 210 is connected to process monitor and/or TCP/IP Stack in order to monitor running processes and traffic flow. The status of various systems being monitored is transmitted to the database and decision engine 220.

20

The database and decision engine 220 processes data received from the monitoring engine 210 and performs the required processes in accordance to the data received. The database and decision engine 220 receives data from monitoring engine 210 and makes decision to either update database or provide instructions to controller 230 to pre-fetch DNS or content, and perform other pre-emptive operations.

25

The controller 230 receives instructions from the database and decision engine and executes the instructions accordingly.

In an embodiment, the enhancer module 205 is installed in a client operating system. In another embodiment, the enhancer module 205 is installed in a router/server. In yet another embodiment, where the client operating system is connected to a server or router, the enhancer module 205 may be installed in both client operating system and server/router as shown in figure 2.

The programming languages that may be used for writing the programme in accordance with embodiments of this invention include Scala, Java and C languages. Nevertheless, one skilled in the art will recognise that programming languages may be used without departing from the invention. The processes in accordance with embodiments of this invention are executable on the various platforms such as Windows, Linux, Mac, and Android. Nevertheless, one skilled in the art will recognise that the processes in accordance with embodiments of this invention may also be executable of other platforms without departing from this invention.

The profiles of all users are stored on the database. The profiles include network interface hardware address, MAC address of default gateway, MAC address of DHCP server, reachability of network printer, reachability of DNS server(s), number of Internet Protocol (IP) hops to reach selected destination(s), Login name, SSID, cellular network status, etc. The profiles are used for identifying the user or group of users. For each profile, the system associates a number of events commonly executed by the profile. The events may be executing email account, executing a browser, assessing certain websites. Table 1 below illustrates an example of a profile data structure while table 2 illustrates an event data structure for 6 profiles.

Profile	Default Gateway MAC Address	Login name	SSID
Profile 1			
Profile 2			
Profile 3			
Profile 4			
Profile 5			

Table 1

Event	Action 1	Action 2	Action 3	Action 4	Action 5	Action 6
Profile 1	Email account, 10	Powerpoint, 8	Browser, 7	Website 1, 4	Website 2, 3	Website 4, 7
Profile 2	Browser, 8	Website 3, 7	Website 4, 10	Website 2, 4	Website 1, 5	Website 7, 12
Profile 3	Browser, 9	Website 2, 10	Website 1, 8	Website 3, 9	Website 4, 7	Website 7, 5
Profile 4	Email account, 20	Browser, 18	Website 1, 19	Website 2, 12	Website 4, 10	Website 3, 15
Profile 5	Browser, 18	Website 5, 19	Website 6, 11	Website 7, 12	Website 1, 4	Website 2, 3
Profile 6	Email account, 16	Browser, 12	Website 3, 15	Website 4, 16	Website 5, 6	Website 7, 12

Table 2

5

As it is likely that a user performs the same event about the same time of the day, the events may be grouped according to time interval instead of exact time of the day. Hence, the event data structure in table 2 may be extended to specific time period and even day-of-week. Table 3 below illustrates the data structure for events with a time

10 interval of 1 hour between 0900hr to 1300hrs.

Event	Action 1	Action 2	Action 3	Action 4	Action 5	Action 6
0900-1000, profile 1	Email account, 10	Powerpoint, 8	Browser, 7	Website 1, 4	Website 2, 3	Website 4, 7
0900-1000, Profile 2	Browser, 8	Website 3, 7	Website 4, 10	Website 2, 4	Website 1, 5	Website 7, 12
0900-1000, Profile 3	Browser, 9	Website 2, 10	Website 1, 8	Website 3, 9	Website 4, 7	Website 7, 5
1100-1200, Profile 1	Browser, 7	Website 1, 4	Website 2, 3	Website 5, 16	Website 5, 6	Website 7, 12
1200-1300,	Browser,	Website 3,	Website	Website	Website	Website

Profile 2	8	7	4, 10	1, 5	1, 8	3, 9
1200-1300, Profile 3	Browser, 9	Website 2, 10	Website 1, 8	Website 1, 4	Website 2, 3	Website 4, 10

Table 3

One skilled in the art will recognise that the time interval needs not be restricted to one hour and may be shorter or longer. Further the time interval may vary adaptively depending on the time-of-day and/or day-of-week and/or day-of-month. For example, for peak period (e.g. lunch time, end of working day, weekdays, first day of month, etc), the time interval can be 5 minutes. During non-peak period, time interval may then be changed to 1 hour or longer. Hence, other time intervals may be implemented without departing from the invention. The exact time interval is a design choice left to those skilled in the art.

Tables 2 and 3 illustrate an event data structure comprising an event list for each profile and each event list includes 6 actions. One skilled in the art will recognise that the number of actions need not be restricted to 6 actions and that any other number of actions may be implemented without departing from the invention.

Each action is accompanied by a counter showing the number of times being executed or triggered by the profile. In each event list, the actions are grouped into 2 groups, namely, the active group and the reserve group. The active group comprises top few actions that are commonly triggered by the profile. The reserve group comprises actions that are either newly triggered or less commonly triggered by the profile. Essentially, the active group is updated with actions that are more commonly triggered than the reserve group.

A method of enhancing the DNS performance in accordance with embodiments of the invention will now be described.

Figure 3 illustrates a flow diagram of process 300 performed by the enhancer 205 being installed on a client operating system or on a router/server in accordance with an embodiment of this invention. Process 300 begins with step 305 by determining the profile. The profile may be determined in the following manner when the enhancer 205 is installed on a client operating system. The monitoring engine retrieves userid in the client Operating System (OS). The userid is stored in the client operating system when the user logs in to the operating system. The database & decision engine then uses the userid and other parameters retrieved by the monitoring engine to determine the profile based on the profile data structure. One skilled in the art will recognise that if userid is not available, other parameters such as MAC address of DHCP server and reachability of DNS server(s) can also be used for profile determination. As for an enhancer 205 being installed on a router/server, the profile may be determined in the following manner. The monitoring engine retrieves the MAC address of client. The database & decision engine then uses the MAC address and other parameters retrieved by the monitoring engine to determine the profile based on the profile data structure. For brevity, we have used userid and MAC address of client as parameters for determining a profile of the user. As mentioned above in table 1, other possible parameters such as "SSID" and "Default DNS Server IP Address", etc. may also be used for determining a profile of the user. Further, the determination of the profile may be a combination of a few parameters.

In step 310, process 300 retrieves all the events associated to the determined profile. Particularly, the database & decision engine retrieves the events, from the event data structure, associated to the determined profile.

In step 315, process 300 clears the probability score and proceeds to step 320 to monitor for an event taken by the user. Upon detecting an event, process 300 performs a

match to determine if the detected event matches any of the actions in the active group in the event list associated to the profile in step 325. If the detected event matches any of the actions in the active group in event list associated to the profile, process 300 proceeds to step 330. If the detected event does not match with any of the actions in the
5 active group in event list associated to the profile, process 300 proceeds to step 350.

In step 350, process 300 performs a match to determine if the detected event matches any of the actions in the reserve group in the event list associated to the profile. If the detected event matches any of the actions in the reserve group, process 300
10 proceeds to step 360. If the detected event does not match with any of the actions in the reserve group, process 300 proceeds to step 355 to add a new entry. Particularly, the detected event is added as a new action in the reserve group in step 355.

In step 360, process 300 increments the counter of the matched action in the
15 reserve group. Process 300 then determines if the counter of the matched action is more than any of the counters in the active group in step 365. If the counter of the matched action is more than any of the counters in the active group, process 300 proceeds to step 370 to update the active group by replacing the action, having a counter that is lesser than the counter of the matched action, with the matched action. If the counter of the
20 matched action is not more than any of the counters in the active group, process 300 repeats from step 320 to monitor for the next event.

In step 370, process 300 updates the active group in the following manner. The action in the active group with lesser counter is being replaced by the matched action.
25 Subsequently, the weightage among the actions in the active group is being apportioned according to user requirements. The weightage may be apportioned according to the value of the counters of each action in the active group or evenly distributed. Alternatively,

recent action being executed may be accorded with higher weightage. One skilled in the art will recognise that other methods of apportioning the weightage may be implemented without departing from the invention.

5 In step 330, process 300 increments the counter of the matched action. Process 300 then calculate a score in step 335 and append the score to the probability score. The score is calculated in the following manner. A weightage is provided for each of the actions in the active group. For example, if an event list comprises 10 actions in the active group, each action will be accorded 0.1. Assuming Action 2 is being triggered first, the
10 score would be $(0.1)*100\% = 10\%$. If subsequently Actions 4, 6, 7, 8 and 9 are detected, the score would be $(0.1+0.1+0.1+0.1+0.1+0.1)*100\% = 60\%$. This example illustrates apportioning the weightage evenly among the actions in the active group. However, one skilled in the art will recognise that the weightage may be apportioned according to the counter of each of the actions or any other method without departing from the invention.

15 In step 340, process 300 determines whether the score is above a threshold of 50%. If the score is above the threshold of 50%, process 300 proceeds to step 345. If the score is not above the threshold of 50%, process 300 repeats from step 320 to monitor for the next event. One skilled in the art will recognise that the threshold value may vary
20 accordingly and the exact threshold value is left to those skilled in the art.

 In step 345, process 300 executes pre-emptive operations associated to the predicted actions. In one embodiment, the pre-emptive operations are made up by the remaining actions in the event list of associated to the profile. In another embodiment, a
25 list of pre-emptive actions can also be associated with each profile, and the pre-emptive actions will be executed by the Controller. The lists and associated actions can be pre-defined by administrators, or periodically self-generated by the Enhancers employing

various Machine Learning mechanisms. Table 4 below illustrates an event data structure with pre-emptive actions being listed for each of the two profiles.

Event	Action 1	Action 2	Action 3	Action 4	Action 5	Pre-emptive Action 1	Pre-emptive Action 2
0900-1000, Profile 1	Email account	powerpoint	browser	Website 1	Website 2	Check Website 1 SSL certificate	Perform DNS query to last known DNS Server
0900-1000, Profile 2	Email account	browser	Website 1	Website 2		Check Website 3 reputation	

Table 4

5

Some examples of pre-emptive operations are as follows:

1. performs a DNS query in order to update DNS cache.
2. performs pre-emptive network events such as connecting to email server.
3. send DNS query to last used DNS server(s) and use the first DNS reply.
4. initiate check of server and/or client SSL certificate and initiate check of server's Internet reputation.

10

Process 300 ends after step 345.

15

Figure 4 illustrates a flow diagram of process 400 performed by the enhancer 205 being installed on a client operating system and possibly on a router/server in accordance with an embodiment of this invention. Process 400 illustrates a process to decide on which of the two enhancers (i.e. client or router/server) to perform process 300. For brevity, we indicate (client) and (router/server) to differentiate between the enhancer 205 on client operating system and the enhancer 205 on router/server respectively. Process 400 begins with step 405 by transmitting a pre-emptive operations request in the following manner. The pre-emptive operations request is generated by the database & decision

20

engine (client) and transmitted by the controller (client) to the controller (router/server).
The pre-emptive operations request includes the profile.

In step 410, the enhancer (client) monitors for an acknowledgement from the
enhancer (router/server). If the enhancer (client) receives an acknowledgement from the
enhancer (router/server), process 300 will be executed by the enhancer (router/server) in
step 415. If the enhancer (client) does not receive an acknowledgement from the
enhancer (router/server) within a certain period of time, process 300 will be executed by
the enhancer (client) in step 420.

Figure 5 illustrates a timing diagram of process 500 performed by the enhancer
205 being installed on a client operating system and on a router/server in accordance
with an embodiment of this invention. Process 500 illustrates a process to combine the
scores calculated by respective enhancers to determine executing pre-emptive
operations. The left side of figure 5 illustrates the process undertaken by the enhancer
(client) while the right side of figure 5 illustrates the process undertaken by the enhancer
(router/server). Dashed lines between enhancer (client) and enhancer (router/server)
illustrates transmission of data between enhancer (client) and enhancer (router/server).

Process 500 begins with 505 by transmitting a pre-emptive operations request to
the enhancer (router/server). Upon receipt of the pre-emptive operations request in step
605, the enhancer (router/server) generates and transmits an acknowledgement to the
enhancer (client) in step 610. In step 510, enhancer (client) receives the
acknowledgement from the enhancer (router/server). This establishes a connection
between the enhancers.

After step 610, the enhancer (router/server) performs process 300 as usual with an added step between step 335 and step 340. In particular, the score is transmitted to the enhancer (client).

5 After step 510, the enhancer (client) performs process 300 with a modification to the calculation of the score in step 335. In particular, the score is calculated factoring in the score from the enhancer (router/server). The score is calculated in the following manner.

10 The total weightage of enhancer (client) and enhancer (router/server) is 1. In this example, the enhancer (client) weightage is 0.6 while the enhancer (router/server) is 0.4. One skilled in the art will recognise that weightage of both enhancers may be equally apportioned or vary without departing from the invention. Below table 5 illustrates the calculation of the score for the events detected by the enhancer (client) and table 6 illustrates the calculation of the score for the events detected by the enhancer (router/server). In this instance, the events detected are “check email”, “powerpoint”, and “website 2”.

Event	Client (e.g. Client Weight =0.6)					Profile/Event Probability
	Action 1	Action 2	Action 3	Action 4	Action 5	
Profile 1	Check Email, 30 (weightage = 0.3)	Powerpoint, 10 (weightage = 0.1)	Website 1, 25 (weightage = 0.25)	Website 2, 10 (weightage = 0.1)	Website 4, 25 (weightage = 0.25)	$0.6 \times (0.3 + 0.1 + 0.1) = 0.3$

Table 5

Event	Router/Server (e.g. Server Weight = 0.4)					Profile/Event Probability
	Action 1	Action 2	Action 3	Action 4	Action 5	
Profile 1	Check Email, 25 (weightage = 0.25)	Powerpoint, 25 (weightage = 0.25)	Website 1, 20 (weightage = 0.2)	Website 4, 20 (weightage = 0.2)	Website 8, 10 (weightage = 0.1)	$0.4 \times (0.25 + 0.25) = 0.2$

Table 6

The score (Net Probability) calculated by the enhancer (client) would be $(0.3 + 0.2) \times 100\% = 50\%$.

- 5 In addition, the Net Probability can also include a weighted Probability of an Action for past few days, or same day (e.g. Mondays) for past 1 month. The weights for more recent weeks will be higher. For example, the Client Weight for current week (i.e. Week 1) can be 1/2, Client Weight for Week 2 is 1/4, Client Weight for Week 3 is 1/8, etc. Essentially, more weightage is apportioned to more recent data. See table 7 below for
- 10 illustration.

Event	Client (e.g. Weight =0.6)					Server (e.g. Weight =0.4)				Profile/Event Probability
	Action 1	Action 2	Action 3	Action 4	Action 5	Action 1	Action 2	Action 3	Action 4	
Week 1	Client Weight for Week 1 =0.6*1/2					Server Weight for Week 1 = 0.4*1/2				
Monday, Profile 1	Check Email, 30 (weightage = 0.3)	Powerpoint, 10 (weightage = 0.1)	Website 1, 25 (weightage = 0.25)	Website 2, 10 (weightage = 0.1)	Website 4, 25 (weightage = 0.25)	Check Email, 25 (weightage = 0.25)	Powerpoint, 25 (weightage = 0.25)	Website 1, 25 (weightage = 0.25)	Website 4, 25 (weightage = 0.25)	$((0.6/2) \times (0.3 + 0.1 + 0.1))$ $+$ $((0.4/2) \times (0.25 + 0.25)) = 0.25$
Week 2	Client Weight for Week 2 =0.6*1/4					Server Weight for Week 2 = 0.4*1/4				
Monday, Profile 1	Check Email, 30 (weightage = 0.3)	Powerpoint, 10 (weightage = 0.1)	Website 1, 25 (weightage = 0.25)	Website 2, 10 (weightage = 0.1)	Website 4, 25 (weightage = 0.25)	Check Email, 25 (weightage = 0.25)	Powerpoint, 25 (weightage = 0.25)	Website 1, 25 (weightage = 0.25)	Website 4, 25 (weightage = 0.25)	$((0.6/4) \times (0.3 + 0.1 + 0.1))$ $+$ $((0.4/4) \times (0.25 + 0.25)) = 0.125$
Week 3	Client Weight for Week 3 =0.6*1/8					Server Weight for Week 3 = 0.4*1/8				
Monday, Profile 1	Check Email, 30 (weightage = 0.3)	Powerpoint, 10 (weightage = 0.1)	Website 1, 25 (weightage = 0.25)	Website 2, 10 (weightage = 0.1)	Website 4, 25 (weightage = 0.25)	Check Email, 25 (weightage = 0.25)	Powerpoint, 25 (weightage = 0.25)	Website 1, 25 (weightage = 0.25)	Website 4, 25 (weightage = 0.25)	$((0.6/8) \times (0.3 + 0.1 + 0.1))$ $+$

	htage = 0.3)	= 0.1)	= 0.25)	= 0.1)	e = 0.25)	0.25)	= 0.25)	= 0..25)	= 0.25)	$((0.4/8)*$ $(0.25 +$ $0.25) =$ 0.0625
--	--------------------	-----------	------------	-----------	--------------	-------	------------	-------------	------------	---

Table 7

Process 500 illustrates enhancer (router/server) transmitting score to the enhancer (client) and the net probability score is being calculated by the enhancer (client).

5 One skilled in the art will recognise that process 500 may be modified such that the enhancer (client) transmits its score to the enhancer (router/server) and let the enhancer (router/server) compute the net probability score instead. Further, the score from both enhancers may be transmitted to each other so that net probability may be computed at both ends. In accordance with an embodiment of the invention, a user profile may be

10 linked with group profile(s). For example, profiles 1, 4 and 5 belong to a "HR" group as these profiles are from the HR department and profiles 2, 3 and 6 belongs to a "Finance" group as these profiles are from the Finance department. With these groupings, the enhancer (router/server) can also include the probability of group profile(s) in determining the Net probability score. The group profile(s) can be determined by the enhancer

15 (server/client) by querying existing enterprise LDAP databases such as Windows Active Directory. In another embodiment, the enhancer (server/router) can calculate group profile(s) for all the existing users via machine learning algorithms such as Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) and k-Means.

20 The above is a description of exemplary embodiments of a system and method in accordance with this invention. It is foreseeable that those skilled in the art can and will design alternative systems based on this disclosure that infringe upon this invention as set forth in the following claims.

Claims:

1. A system for performing pre-emptive operations comprising:

an enhancer installed on a client operating system comprises,

5 a monitoring engine communicatively connected to TCP/IP stack and processes,

a database and decision engine having a database storing a profile data structure and an event data structure,

a controller executing instructions received from database and decision engine,

10 instructions stored on said database to perform the following steps:

determining a profile of a user based on said profile data structure,

retrieving an event list from said event data structure associated to said determined profile,

monitoring an event,

15 computing a probability score for said event,

determining whether said probability score is above a predetermined threshold; and

executing pre-emptive operations in response to said probability score being above said predetermined threshold, wherein said event list in said event data structure is

20 divided into two groups, a first group being an active group and a second group being a reserve group and said instruction to monitoring said event further comprises:

clearing a probability score on said database;

detecting an event;

matching said event to one of a plurality of actions in said active group;

25 increment a counter of said one of a plurality of actions in said active group in response to said event matches one of said plurality of actions in said active group; and

computing said probability score for said event.

2. The system for performing pre-emptive operations according to claim 1, wherein said instruction to determining the profile of a user based on said profile data structure comprises:

retrieving a parameter stored on a client operating system;

matching said parameter with parameters in said profile data structure; and

determining one of a plurality of profiles in said profile data structure in response to said parameter matching a parameter of said one of said plurality of profiles.

3. The system for performing pre-emptive operations according to claim 1, wherein said instruction to computing said probability score for said event comprises:

retrieving a weightage of said one of said plurality of actions; and

adding said weightage to said probability score.

4. The system for performing pre-emptive operations according to claim 1, wherein said instruction to monitoring an event further comprises:

matching said event to one of a plurality of actions in said reserve group in response to said event not being able to match with any of said plurality of actions in said active group; and

adding a new entry in said reserve group in response to said event not being able to match with any of said plurality of actions in said reserve group.

5. The system for performing pre-emptive operations according to claim 4, wherein said instruction to monitoring an event further comprises:

increment a counter of said one of said plurality of actions in said reserve group in response to said event matches one of said plurality of actions in said reserve group;

determining whether said counter of said one of said plurality of actions in said reserve group is more than any one of said plurality of actions in said active group;

updating said plurality of actions in said active group in response to said counter of said one of said plurality of actions in said reserve group being more than any one of
5 said plurality of actions in said active group.

6. The system for performing pre-emptive operations according to claim 5, wherein said instruction to updating said plurality of actions in said active group further comprises:

10 replacing one of said plurality of actions in said active group with said one of said plurality of actions in said reserve group with a counter being more than said one of said plurality of actions in said active group; and

apportioning a weightage among said plurality of actions in said active group.

15 7. The system for performing pre-emptive operations according to claim 6, wherein said instruction to apportioning a weightage among said plurality of actions in said active group further comprises:

apportioning said weightage according to the value of the counters of each action in said active group.

20

8. The system for performing pre-emptive operations according to claim 3 further comprises a second enhancer installed on a server communicatively connected to said client operating system, said second enhancer comprising:

a monitoring engine communicatively connected to TCP/IP stack of said server,

25 a database and decision engine having a database storing a profile data structure and an event data structure,

a controller executing instructions received from database and decision engine,
and

instructions stored on said database to perform the following steps:

determining a profile of a user based on said profile data structure,

5 retrieving an event list from said event data structure associated to said
determined profile,

monitoring an event,

computing a probability score for said event,

transmitting said probability score to said enhancer installed on said client
10 operating system.

9. The system for performing pre-emptive operations according to claim 8
wherein said instruction to computing a probability score for said event stored on said
enhancer installed on said client operating system comprises:

15 retrieving a weightage for said probability score;

applying said weightage of said probability score to said probability score to
determine a first score;

retrieving a weightage of said probability score of said server;

applying said weightage of said probability score of said server to said probability
20 score received from said second enhancer to determine a second score;

adding said first and second scores to determine a net probability score; and

executing said pre-emptive operations in response to said net probability score
being above said predetermined threshold.

25