ATTORNEY DOCKET NO.: CIS03-20(0000)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): Philip J.S. Gladstone and Jeffrey A. Kraemer
Serial No.: 10/172,305
For: Stateful Distributed Event Processing And Adaptive Security
Filing Date: June 14, 2002
Examiner: Mohammad W. Reza
Art Unit: 2136
Conf. No.: 3007

Certificate of Transmission Under 37 C.F.R. §1.8

I hereby certify that this correspondence is being electronically transmitted to the United States Patent and Trademark Office via the EFS-Web system on: June 13, 2007

Date: October 18, 2007

By: Penny A. Coelho
(Typed or printed name of person mailing Document, whose signature appears below)

Signature: /pac/

MAIL STOP AF
Commissioner for Patents
PO Box 1450
Alexandria, Virginia 22313

"Do not Entered"
MR
11/06/07

Sir:

AMENDMENT

In response to the Final Office Action mailed on August 21, 2007, please amend the above-identified Application as follows:

ATTORNEY DOCKET NO.: CIS03-20(0000)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): Philip J.S. Gladstone and Jeffrey A. Kraemer
Serial No.: 10/172,305
For: Stateful Distributed Event Processing And Adaptive Security
Filing Date: June 14, 2002
Examiner: Mohammad W. Reza
Art Unit: 2136
Conf. No.: 3007

Certificate of Transmission Under 37 C.F.R. §1.8

I hereby certify that this correspondence is being electronically transmitted to the United States Patent and Trademark Office via the EFS-Web system on: June 13, 2007

Date: October 18, 2007

By: Penny A. Coelho
(Typed or printed name of person mailing Document, whose signature appears below)

Signature: /pac/

MAIL STOP AF
Commissioner for Patents
PO Box 1450
Alexandria, Virginia 22313

Sir:

AMENDMENT

In response to the Final Office Action mailed on August 21, 2007, please amend the above-identified Application as follows:

IN THE CLAIMS

1. (Currently Amended) A method of maintaining a networked computer system including first and second nodes and an event processing server, comprising:

the first and second nodes detecting changes in state;

the event processing server receiving notification of the changes in state from the first and second nodes;

the event processing server correlating the changes in state detected by the first and second nodes;

the event processing server executing a maintenance decision which affects the first and second nodes, wherein the maintenance decision is based on the correlating of the changes in state detected by the first and second nodes, the changes in state a result of an absence of an event;

wherein the absence of an event comprises at least one of:

an absence of a request for system resources; and

an absence of an event message received within a predetermined time frame; ~~and~~

wherein the detecting, transmitting, correlating, and executing occurs without human intervention[[.]]; and

wherein the event processing server comprises an interceptor inserted in a communication path of the networked computer system, the method further comprising:

at the interceptor, detecting an access request in the communications path;

generating an event message for the access request;

transmitting the event message to the event processing server; and

in response, receiving a policy message from the event processing server comprising at least one of:

instructions for allowing the access request to continue along the communications path, and

instructions for disallowing the access request to continue along the

communications path.

2. (Canceled)

3. (Original) The method of claim 1 wherein the changes in state are recognized by a reference monitor.

4. (Original) The method of claim 3 wherein the monitor is a stateful reference monitor.

5. (Original) The method of claim 1 wherein the event processing server receiving the report is the result of one of the first and second nodes reporting to the event processing server, and the event processing server polling the first and second nodes.

6. (Original) The method of claim 1 further including the event processing server updating an operating policy on the network.

7. (Original) The method of claim 6 wherein the updating the operating policy includes at least one of requesting security policy changes on at least one node, requesting changes to privileges to access system resources on at least one node, tuning system parameters on at least one node, and modifying network firewall parameters.

8. (Original) The method of claim 6 further including at least one node enacting the updated operating policy.

9. (Original) The method of claim 1 further including notifying an external entity of actions taken.

10. (Original) The method of claim 9, wherein the external entity is a network administrator.

11. (Original) The method of claim 9, wherein the external entity is a software application executing on the network.

12. (Currently Amended) A method for maintaining a networked computer system including:

at least one node detecting at least one change in state;

an event processing server on the network receiving notification of the at least one change in state from the at least one node; and

the event processing server responding to the notification by executing a maintenance decision, wherein the maintenance decision is based on the at least one change in state from the at least one node, the at least one change in state a result of an absence of an event;

wherein the absence of an event comprises at least one of:

an absence of a request for system resources; and

an absence of an event message received within a predetermined time frame; ~~and~~

wherein the detecting, receiving, and responding occurs without human intervention[[.]]; and

wherein the event processing server comprises an interceptor inserted in a communication path of the networked computer system, the method further comprising:

at the interceptor, detecting an access request in the communications path;

generating an event message for the access request;

transmitting the event message to the event processing server; and

in response, receiving a policy message from the event processing server comprising at least one of:

instructions for allowing the access request to continue along the communications path, and

instructions for disallowing the access request to continue along the communications path.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.