(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
18 July 2002 (18.07.2002)

PCT

(10) International Publication Number
WO 02/056156 A2

(51) International Patent Classification⁷: **G06F 1/00**

(21) International Application Number: PCT/GB01/05767

(22) International Filing Date:
24 December 2001 (24.12.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
0100955.4 13 January 2001 (13.01.2001) GB

(71) Applicant *(for all designated States except US)*: **QINETIQ LIMITED** [GB/GB]; 85 Buckingham Gate, London SW1E 6PD (GB).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: **WISEMAN, Simon, Robert** [GB/GB]; QinetiQ Malvern, St Andrews Road, Malvern, Worcestershire WR14 3PS (GB).

(74) Agent: WILLIAMS, Arthur, Wyn; QinetiQ Formalities, A4 Building Ively Road, Farnborough, Hampshire GU14 0LX (GB).

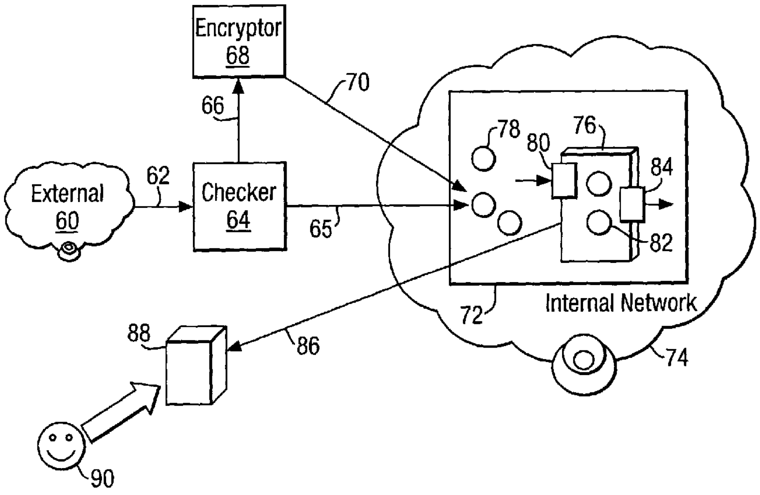
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declaration under Rule 4.17:
— *of inventorship (Rule 4.17(iv)) for US only*

[Continued on next page]

(54) Title: COMPUTER SYSTEM PROTECTION



(57) Abstract: Computer system protection to protect against harmful data from an external computer network (60) (e.g. the Internet) involves supplying incoming data (62) to a software checker (64) as the data enters a computer system (not shown). The checker (64) routes any suspect data (66) to an encryptor (68) which encrypts it to render it unusable and harmless. Encrypted data passes to a computer (72) in an internal network (74) and having a desktop quarantine area or sandbox (76) for suspect data. The computer (72) runs main desktop applications (78) receiving encrypted data (70) for storage and transfer, but not for use in any meaningful way because it is encrypted. Equally well applications (78) cannot be interfered with by encrypted data (70) because encryption makes this impossible. On entry into the sandbox (76), the encrypted data (70) is decrypted to usable form: it then becomes accessible by software (204) suitable for use in the sandbox (76) subject to sandbox constraints.

WO 02/056156 A2

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Computer System Protection

This invention relates to a method for computer system protection against unwanted external interference such as for example by viruses, to a computer program for implementing such protection and to a computer system protected thereby.

Computer software applications offer progressively more flexible features and become better integrated as computer technology develops. Unfortunately this has the effect of increasing the exposure of computer systems to attack: attacks by Trojan Horse software exploit hidden features of applications software running on a victim's computer, ***and*** attacks by viruses result in software introduced by an attacker spreading from one computer to another. Computer system protection therefore becomes progressively more difficult as technology advances. Attacks on computer systems may damage information they hold, leak that information or prevent legitimate computer system users carrying out their work.

Current industry best practice in computer system protection, as described in the text book "Network Security" by Kaufman, Perlman and Speciner, is to apply a software checker to data as it enters a computer system: the checker identifies a potential attack, allowing any data that appears to be an attack to be rejected. Unfortunately, it is very difficult to identify an attack accurately using software checkers and it is often necessary to err on the side of caution. The result is that data that is harmless and perhaps valuable may not be allowed to enter the system.

A computer system which rejects harmless and sometimes valuable data is not a reliable business tool, so to reduce the loss of data it is known to place rejected data in what is referred to as "quarantine": quarantine is a region of computer storage not accessible by normal users and their software applications such as word processors, but instead accessible by computer experts who can inspect rejected data manually and decide whether or not it is harmful. Expert manual inspection of data in quarantine can be much more accurate at detecting an attack than a software checker. Thus a proportion of data that is rejected by an automatic software checker may subsequently be identified as harmless and allowed to enter the computer system.

Manual inspection of quarantined data improves reliability of communication between a computer system and the outside world, but it results in delay which can be significant and it requires costly expert staff to implement it. Moreover, automatic checkers and manual inspection are both prone to failure. In particular, both automatic and manual checks are poor at detecting new and therefore unfamiliar forms of attack. Forms of attack are associated with functionalities available in applications; new forms of attack therefore appear as software applications are further developed. Hence current industry best practice in computer system protection is costly and ineffective, and this situation will not improve.

Another prior art technique referred to as "sandboxing" is described in the text book "JAVA Security" by Scott Oaks: it provides an alternative to the data rejection approach. In this technique data is allowed to enter a computer system, but the system environment, i.e. the way in which the data can be used, is constrained. Should data prove to constitute an attack, the Trojan Horse or virus which it implements has access only to the constrained environment and cannot corrupt software applications outside it, i.e. beyond the sandbox boundary.

The most common form of sandbox is that provided for JAVA® applets, which are self contained elements of software written in Sun Microsystems' language JAVA that can be executed on a wide variety of different types of computer. Unfortunately, the JAVA® sandbox suffers from the drawback of only working for JAVA® applets and not for data in any other form. For example, a Microsoft® Word document cannot be edited by the Microsoft® Word application within a JAVA® sandbox.

More general-purpose sandboxes have been built or proposed, but are not in general use: examples include research software from University California Berkeley called Janus and described in a paper entitled "Janus: An Approach for Confinement of Untrusted Applications", David A. Wagner, UC Berkeley Computer Science Division, report CSD-99-1056, August 1999. These utilise security features within an operating system to separate software executing within the sandbox from other software executing on a computer system in the form of a main workstation desktop.

The use of sandboxing does not, however, really solve the problem. This is because viruses may still spread freely within the constrained environment provided by the sandbox and users will inevitably need to move data across the sandbox boundary, to reflect business needs to exchange data.

It is an object of the invention to provide an alternative form of computer system protection.

The present invention provides computer system protection including a sandbox application for receiving potentially harmful data and defining a sandbox desktop, characterised in that it also includes means for encrypting potentially harmful data to render it harmless and means for decrypting encrypted data for processing by means of an application constrained by the sandbox application.

The invention provides the advantage of enabling potentially harmful data to be examined and executed while constrained by the sandbox application: this in turn allows a user to decide the data's importance while the data is quarantined by encryption. Unwanted material can be discarded, avoiding the need for further inspection. Moreover, important messages need not be delayed awaiting expert inspection, but instead made available to a system user in a constrained quarantine environment provided by a sandbox desktop.

The sandbox application may be arranged to employ a desktop application which does not communicate with applications associated with a main desktop of a computer system to which the protection is applied.

The computer system protection may includes means for enabling a user to retrieve data from the sandbox application in encrypted form for relaying to expert inspection and means for checking decrypted data released from the sandbox desktop for potentially harmful content. It may be mounted upon a computer linked via a firewall to an external network.

In another aspect, the invention provides a protected computer system having a sandbox application for receiving potentially harmful data and defining a sandbox desktop, characterised in that it also includes a firewall protecting a checker from an external

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.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE