



(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẢNG ĐỘC QUYỀN SÁNG CHẾ

(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN)
CỤC SỞ HỮU TRÍ TUỆ

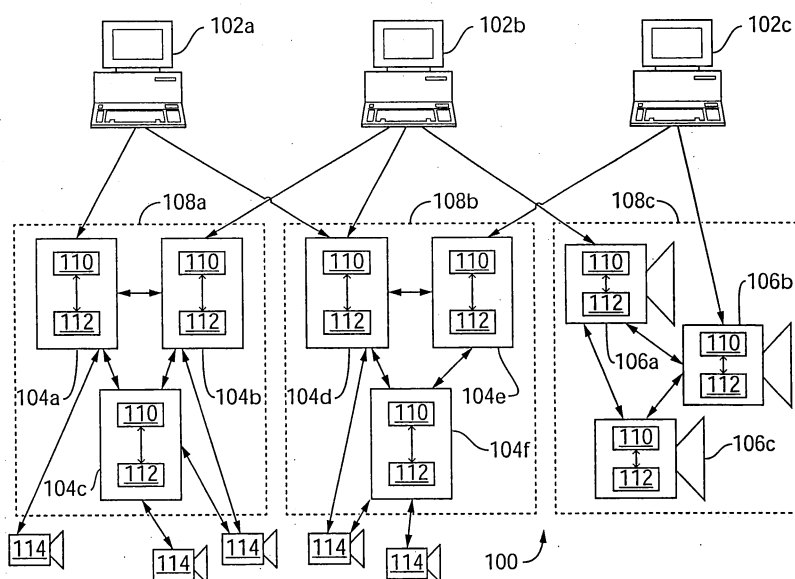
(11) 
1-0021499

(51)⁷ H04L 12/16, G08B 13/00, H04L 12/24, (13) B
H04N 7/18

- (21) 1-2015-01039 (22) 06.09.2013
(86) PCT/CA2013/050690 06.09.2013 (87) WO2014/036656 13.03.2014
(30) 13/607,447 07.09.2012 US
(45) 26.08.2019 377 (43) 27.07.2015 328
(73) AVIGILON CORPORATION (CA)
Box 378, #101 - 1001 West Broadway, Vancouver, BC V6H 4E4, Canada
(72) LEE, Ryan (CA), MARLATT, Shaun (CA), ADAM, Matthew (CA), WIGHTMAN,
Ross (CA), MAGOLAN, Greg (CA), MARTZ, Andrew (CA)
(74) Công ty Luật TNHH T&G (TGVN)

(54) HỆ THỐNG BẢO MẬT VẬT LÝ VÀ PHƯƠNG PHÁP CHIA SẺ DỮ LIỆU
TRONG HỆ THỐNG BẢO MẬT VẬT LÝ

(57) Sáng chế đề cập đến hệ thống bảo mật vật lý có nhiều nút máy chủ có thể được xây dựng làm mạng phân tán. Để gửi dữ liệu giữa các nút trong mạng, nút thứ nhất có thể truy cập mã nhận dạng nút nhận dạng nút thứ hai, với cả hai nút thứ nhất và thứ hai tạo thành ít nhất một phần của cụm máy chủ, và sau đó, nút thứ nhất có thể gửi dữ liệu đến nút thứ hai. Mã nhận dạng nút tạo thành ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả và có thể truy cập tất cả nút máy chủ trong cụm máy chủ đó. Chức năng như khả năng chia sẻ các hiển thị giữa các người sử dụng của hệ thống và khả năng đối với các người dùng này để điều khiển màn hình không được giám sát có thể được thực hiện trên mạng phân tán, mạng trung ương, hoặc loại mạng khác.



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến hệ thống bảo mật vật lý có nhiều nút máy chủ.

Tình trạng kỹ thuật của sáng chế

Hệ thống bảo mật vật lý là hệ thống thực hiện các biện pháp ngăn ngừa những người không được quyền truy cập vật lý đến tài sản, như tòa nhà, hạ tầng, hoặc thông tin bí mật. Các ví dụ về các hệ thống bảo mật vật lý bao gồm các hệ thống giám sát, như hệ thống mà trong đó các máy quay được sử dụng để kiểm soát tài sản và các thứ gần nó; các hệ thống điều khiển truy cập, như hệ thống sử dụng các thẻ RFID để điều khiển truy cập tòa nhà; các hệ thống phát hiện xâm nhập, như hệ thống cảnh báo ăn trộm tại nhà; và các tổ hợp của các hệ thống trước đây.

Hệ thống bảo mật vật lý thường tích hợp với các máy tính. Do loại hệ thống bảo mật vật lý này phát triển, nên công suất tính toán yêu cầu để vận hành hệ thống tăng lên. Ví dụ, do số máy quay trong hệ giám sát tăng lên, lượng công suất tính toán theo yêu cầu cũng gia tăng để cho phép video bổ sung được lưu trữ và cho phép sử dụng đồng thời và quản lý số máy quay cao hơn. Do đó, việc nghiên cứu và phát triển tiếp tục khắc phục các vấn đề gặp phải do hệ thống bảo mật vật lý phát triển.

Bản chất kỹ thuật của sáng chế

Theo khía cạnh thứ nhất, sáng chế đề xuất phương pháp chia sẻ dữ liệu trong hệ thống bảo mật vật lý mà bao gồm nhiều nút máy chủ. Phương pháp này bao gồm việc truy cập, sử dụng một trong số các nút máy chủ (“nút thứ nhất”), mã nhận dạng nút nhận dạng nút khác của các nút máy chủ (“nút thứ hai”), trong đó nút thứ nhất và thứ hai bao gồm ít nhất một phần của cụm máy chủ và trong đó mã nhận dạng nút bao gồm ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả và có thể truy cập bằng tất cả các nút máy chủ trong cụm máy chủ; và gửi dữ liệu từ nút thứ nhất đến nút thứ hai.

Cụm máy chủ có thể bao gồm ít nhất ba nút máy chủ.

Các nút máy chủ có thể bao gồm máy quay, máy ghi video mạng, và các máy chủ điều khiển truy cập.

Phương pháp còn bao gồm bước truy cập, sử dụng nút thứ hai, mã nhận dạng nút nhận dạng nút thứ nhất; và gửi thông tin bổ sung từ nút thứ hai đến nút thứ nhất.

Thông tin hội viên của cụm có thể bao gồm mã nhận dạng nút nhận dạng duy nhất từng nút trong số các nút máy chủ trong cụm máy chủ; và mã nhận dạng cụm nhận dạng duy nhất cụm máy chủ mà các nút máy chủ thuộc về đó.

Mỗi nút trong số các nút máy chủ trong cụm máy chủ có lưu trữ lâu dài riêng phiên bản của nó về thông tin hội viên của cụm cục bộ.

Phương pháp còn bao gồm bước khởi động lại một trong số các nút máy chủ (“nút máy chủ được khởi động lại”) trong cụm máy chủ; và mỗi lần nút máy chủ được khởi động lại trở lại trực tuyến, thì sử dụng nút máy chủ được khởi động lại để thực hiện phương pháp bao gồm (i) truy cập mã nhận dạng cụm nhận dạng cụm máy chủ; và (ii) tự động quay lại cụm máy chủ.

Phương pháp còn bao gồm bước bổ sung nút máy chủ mới vào cụm máy chủ bằng cách thực hiện phương pháp bao gồm bước trao đổi phiên bản của thông tin hội viên của cụm được lưu trữ trên nút máy chủ mới bằng phiên bản của thông tin hội viên của cụm được lưu trữ trên một trong số các nút máy chủ mà đã là một phần của cụm máy chủ (“nút điều khiển hội viên”); và đồng bộ hóa các phiên bản của thông tin hội viên của cụm được lưu trữ trên nút máy chủ mới bằng các phiên bản của thông tin hội viên của cụm được lưu trữ trên tất cả các nút máy chủ trong cụm trước khi nút máy chủ mới tham gia vào nhóm.

Việc gửi dữ liệu có thể bao gồm tìm kiếm, sử dụng nút thứ nhất, điểm cuối giao tiếp cho nút thứ hai từ mã nhận dạng nút; và gửi dữ liệu từ nút thứ nhất đến điểm cuối giao tiếp.

Điểm cuối giao tiếp và mã nhận dạng nút có thể bao gồm các mục nhập trong bản đồ mạng liên quan đến các mã nhận dạng nút cho tất cả các nút máy chủ trong cụm máy chủ đến các điểm cuối giao tiếp tương ứng, và mỗi nút trong số các nút máy

chủ trong cụm máy chủ có thể lưu trữ theo cách lâu dài riêng phiên bản của nó về bản đồ mạng cục bộ.

Bản đồ mạng có thể cho phép từng nút trong số các nút máy chủ trong cụm máy chủ gửi dữ liệu đến một nút bất kỳ khác trong số các nút máy chủ trong cụm máy chủ mà không sử dụng máy chủ được tập trung.

Dữ liệu có thể được lưu trữ cục bộ trên nút thứ nhất và phương pháp còn bao gồm việc biến đổi dữ liệu sử dụng nút thứ nhất, trong đó việc gửi dữ liệu từ nút thứ nhất đến nút thứ hai bao gồm phần đồng bộ hóa dữ liệu trên các nút thứ nhất và thứ hai sau khi nút thứ nhất biến đổi dữ liệu.

Dữ liệu có thể bao gồm thông tin phiên bản được tạo ra bằng cách sử dụng cơ chế tạo phiên bản mang tính nhân quả và các phiên bản khác nhau của dữ liệu có thể được lưu trữ trên các nút thứ nhất và thứ hai, và đồng bộ hóa dữ liệu có thể bao gồm việc so sánh thông tin phiên bản được lưu trữ trên nút thứ nhất và thứ hai và chấp nhận trên cả hai nút thứ nhất và nút thứ hai mà dữ liệu có thông tin phiên bản chỉ ra là gần hơn.

Dữ liệu có thể bao gồm mã nhận dạng nút của nút thứ nhất, thông tin trạng thái nhịp của nút thứ nhất, thông tin trạng thái ứng dụng của nút thứ nhất, và thông tin phiên bản, và gửi dữ liệu có thể bao gồm truyền dữ liệu đến tất cả nút máy chủ trong cụm máy chủ sử dụng giao thức tán gẫu (gossip) thực hiện các trao đổi dữ liệu giữa các cặp nút máy chủ trong cụm.

Dữ liệu có thể được truyền theo giai đoạn đến tất cả các nút máy chủ trong cụm máy chủ.

Dữ liệu được gửi đến nút thứ hai khi nút thứ nhất tham gia vào cụm này.

Miền tập trung các mục nhập có thể được biến đổi bằng các nút máy chủ bất kỳ trong cụm máy chủ có thể được lưu trữ cục bộ trên mỗi nút trong cụm máy chủ, và phương pháp còn bao gồm việc tạo thông tin phiên bản sử dụng cơ chế tạo phiên bản mang tính nhân quả sao cho thông tin phiên bản chỉ ra các nút máy chủ nào có một trong số các mục nhập làm biến đổi một cách gần nhất.

Thông tin trạng thái ứng dụng có thể bao gồm thông tin bấm mức đỉnh được tạo ra bằng cách bấm tất cả các mục nhập trong miền.

Phương pháp còn bao gồm việc so sánh, sử dụng nút thứ hai, thông tin bấm mức đỉnh với thông tin bấm mức đỉnh được tạo ra bằng cách bấm phiên bản của miền tương ứng được lưu trữ cục bộ trên nút thứ hai; và nếu các thông tin bấm mức đỉnh khác nhau, thì sự đồng nhất hóa các miền trên nút thứ nhất và thứ hai sử dụng thông tin phiên bản.

Mục nhập trạng thái chỉ có thể được biến đổi bằng nút thứ nhất có thể được lưu trữ cục bộ trên nút thứ nhất, và thông tin phiên bản có thể bao gồm số phiên bản mà nút thứ nhất gia tăng bất kỳ khi nào nó biến đổi mục nhập trạng thái.

Thông tin trạng thái ứng dụng có thể bao gồm cặp thực thể trạng thái bao gồm mã nhận dạng thực thể trạng thái nhận dạng mục nhập trạng thái và số phiên bản.

Phương pháp còn bao gồm việc so sánh, sử dụng nút thứ hai, số phiên bản nhận được từ nút thứ nhất có số phiên bản của mục nhập trạng thái tương ứng được lưu trữ cục bộ trên nút thứ hai; và nếu các số phiên bản khác nhau, cập nhật mục nhập trạng thái được lưu trữ cục bộ trên nút thứ hai có mục nhập trạng thái được lưu trữ cục bộ trên nút thứ nhất.

Cập nhật mục nhập trạng thái có thể bao gồm việc gửi từ nút thứ nhất đến nút thứ hai các mục nhập trạng thái bổ sung được lưu trữ cục bộ trên nút thứ nhất mà được biến đổi đồng thời với mục nhập trạng thái.

Các nút thứ nhất và hai có thể bao gồm ít nhất một phần của nhóm nút máy chủ trong cụm mà nút thứ nhất có thể gửi dữ liệu theo cách theo thứ tự tổng thể đến tất cả các nút máy chủ trong nhóm, và gửi dữ liệu có thể bao gồm nút thứ nhất gửi dữ liệu đến tất cả các nút máy chủ trong nhóm.

Dữ liệu có thể bao gồm dữ liệu không lâu dài được tạo ra trong thời gian chạy của hệ thống bảo mật vật lý.

Dữ liệu cũng có thể bao gồm việc tạo dòng video được tạo dòng từ các nút máy chủ khác trong cụm máy chủ từ nút thứ nhất đến nút thứ hai.

Theo khía cạnh khác, sáng chế đề xuất hệ thống chia sẻ dữ liệu trong hệ thống bảo mật vật lý, hệ thống này bao gồm nhiều nút máy chủ bao gồm nút thứ nhất và nút thứ hai, trong đó nút thứ nhất bao gồm bộ xử lý được nối theo cách giao tiếp với vật ghi có thể đọc được bằng máy tính mà có thể được mã hóa trên đó các câu lệnh và các chỉ dẫn để làm cho bộ xử lý thực hiện phương pháp bao gồm việc truy cập mã nhận dạng nút nhận dạng nút thứ hai, trong đó nút thứ nhất và thứ hai bao gồm ít nhất một phần của cụm máy chủ và trong đó mã nhận dạng nút bao gồm ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả và có thể truy cập bằng tất cả nút máy chủ trong cụm máy chủ; và gửi dữ liệu đến nút thứ hai.

Theo khía cạnh khác, sáng chế đề xuất vật ghi có thể đọc được bằng máy tính không tạm thời mã hóa trên đó các câu lệnh và các chỉ dẫn để làm cho bộ xử lý thực hiện phương pháp chia sẻ dữ liệu trong hệ thống bảo mật vật lý bao gồm nhiều nút máy chủ, phương pháp bao gồm việc truy cập, sử dụng một trong số các nút máy chủ (“nút thứ nhất”), mã nhận dạng nút nhận dạng nút khác trong các nút máy chủ (“nút thứ hai”), trong đó nút thứ nhất và thứ hai bao gồm ít nhất một phần của cụm máy chủ và trong đó mã nhận dạng nút bao gồm ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả và có thể truy cập bằng tất cả nút máy chủ trong cụm máy chủ; và gửi dữ liệu từ nút thứ nhất đến nút thứ hai.

Theo khía cạnh khác, sáng chế đề xuất phương pháp tương tác với màn hình không được giám sát trong hệ thống bảo mật vật lý mà bao gồm nhiều nút máy chủ, phương pháp bao gồm việc gửi, từ một trong số các nút máy chủ (“nút thứ hai”) giao tiếp với màn hình không được giám sát đến nút khác của các nút máy chủ (“nút thứ nhất”) mà giao tiếp với màn hình máy khách, dữ liệu trạng thái hiển thị chỉ định sự hiển thị được hiển thị trên màn hình không được giám sát; và hiển thị, trên màn hình máy khách, ít nhất một phần của hiện thị được hiển thị trên màn hình không được giám sát. Theo một khía cạnh, không có các nút máy chủ nào là máy chủ cổng kết nối được tập trung; theo khía cạnh thay thế, ít nhất một trong số các nút máy chủ là máy chủ cổng kết nối được tập trung.

Phương pháp còn bao gồm việc gửi, từ nút thứ nhất đến nút thứ hai, tin nhắn để thay đổi hiển thị của màn hình không được giám sát; và cập nhật màn hình không được giám sát theo tin nhắn được gửi từ nút thứ nhất đến nút thứ hai.

Nút thứ nhất và thứ hai và ít nhất một trong nhiều nút máy chủ khác có thể bao gồm cụm máy chủ, nút thứ nhất và thứ hai có thể bao gồm ít nhất một phần của nhóm nút máy chủ trong cụm mà nút thứ hai có thể gửi dữ liệu trạng thái hiển thị theo cách theo thứ tự tổng thể đến tất cả các nút máy chủ khác trong nhóm, và gửi dữ liệu trạng thái hiển thị có thể bao gồm nút thứ hai gửi dữ liệu đến tất cả các nút máy chủ khác trong nhóm.

Nút thứ nhất và thứ hai và ít nhất một trong nhiều nút máy chủ khác có thể bao gồm cụm máy chủ, nút thứ nhất và thứ hai có thể bao gồm ít nhất một phần của nhóm nút máy chủ trong cụm mà nút thứ nhất có thể gửi tin nhắn để thay đổi trạng thái của màn hình không được giám sát theo cách theo thứ tự tổng thể đến tất cả các nút máy chủ khác trong nhóm, và nút thứ nhất có thể gửi tin nhắn để thay đổi trạng thái của màn hình không được giám sát đến tất cả các nút máy chủ khác trong nhóm.

Phương pháp còn bao gồm việc gửi từ nút thứ hai đến nút thứ nhất thông báo mà sự hiển thị của màn hình không được giám sát sẵn sàng được điều khiển.

Việc gửi thông báo có thể bao gồm việc truyền thông báo đến tất cả nút máy chủ trong cụm máy chủ sử dụng giao thức tán gẫu thực hiện các trao đổi dữ liệu giữa các cặp nút máy chủ trong cụm.

Trước khi gửi trạng thái của màn hình không được giám sát đến màn hình điều khiển, phương pháp này có thể bao gồm việc truy cập, sử dụng nút thứ hai, mã nhận dạng nút nhận dạng nút thứ nhất, trong đó nút thứ nhất và thứ hai bao gồm ít nhất một phần của cụm máy chủ và trong đó mã nhận dạng nút bao gồm ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả và có thể truy cập bằng tất cả nút máy chủ trong cụm máy chủ.

Thông tin hội viên của cụm có thể bao gồm mã nhận dạng nút nhận dạng duy nhất một trong số các nút máy chủ trong cụm máy chủ; và mã nhận dạng cụm nhận dạng duy nhất cụm máy chủ mà các nút máy chủ thuộc về.

Mỗi nút máy chủ trong cụm máy chủ có thể lưu trữ theo cách lâu dài riêng phiên bản của nó về thông tin hội viên của cụm một cách cục bộ.

Theo khía cạnh khác, sáng chế đề xuất hệ thống bảo mật vật lý bao gồm: màn hình máy khách; màn hình không được giám sát; và nhiều nút máy chủ, trong đó một trong số các nút máy chủ (“nút thứ nhất”) giao tiếp với màn hình máy khách và nút khác trong số các nút máy chủ (“nút thứ hai”) giao tiếp với màn hình không được giám sát, trong đó nút thứ hai được cấu hình để gửi đến nút thứ nhất dữ liệu trạng thái hiển thị chỉ dẫn hiển thị được hiển thị trên màn hình thứ hai và nút thứ nhất được cấu hình để hiển thị, trên màn hình máy khách, ít nhất một phần của hiển thị được hiển thị trên màn hình thứ hai. Theo một khía cạnh, không có các nút máy chủ nào là máy chủ cổng kết nối được tập trung; theo khía cạnh thay thế, ít nhất một trong số các nút máy chủ là máy chủ cổng kết nối được tập trung.

Theo khía cạnh khác, sáng chế đề xuất hệ thống bảo mật vật lý bao gồm: máy khách có màn hình máy khách; màn hình không được giám sát; và nhiều nút máy chủ, trong đó một trong số các nút máy chủ (“nút thứ nhất”) giao tiếp với máy khách và một nút khác trong số các nút máy chủ (“nút thứ hai”) giao tiếp với màn hình không được giám sát, trong đó nút thứ hai được cấu hình để gửi đến nút thứ nhất dữ liệu trạng thái hiển thị chỉ dẫn hiển thị được hiển thị trên màn hình thứ hai và máy khách và nút thứ nhất được cấu hình để hiển thị, trên màn hình máy khách, ít nhất một phần của hiển thị được hiển thị trên màn hình thứ hai. Theo một khía cạnh, không có các nút máy chủ nào là máy chủ cổng kết nối được tập trung; theo khía cạnh thay thế, ít nhất một trong số các nút máy chủ là máy chủ cổng kết nối được tập trung.

Màn hình không được giám sát có thể được kết nối trực tiếp với nút thứ hai hoặc được kết nối gián tiếp với nút thứ hai thông qua, ví dụ, máy khách hoặc trạm làm việc không được giám sát.

Theo khía cạnh khác, sáng chế đề xuất vật ghi có thể đọc được bằng máy tính không tạm thời mã hóa trên đó các câu lệnh và các chỉ dẫn để làm cho bộ xử lý thực hiện phương pháp tương tác với màn hình không được giám sát trong hệ thống bảo mật vật lý mà bao gồm nhiều nút máy chủ, phương pháp bao gồm việc gửi, từ một trong số các nút máy chủ (“nút thứ hai”) giao tiếp với màn hình không được giám sát

đến một nút khác trong số các nút máy chủ (“nút thứ nhất”) giao tiếp với màn hình máy khách, dữ liệu trạng thái hiển thị chỉ dẫn hiển thị được hiển thị trên màn hình không được giám sát; và hiển thị, trên màn hình máy khách, ít nhất một phần của hiển thị được hiển thị trên màn hình không được giám sát.

Theo khía cạnh khác, sáng chế đề xuất phương pháp chia sẻ sự hiển thị (“hiển thị được chia sẻ”) sử dụng hệ thống bảo mật vật lý mà bao gồm nhiều nút máy chủ, phương pháp bao gồm việc: gửi, từ máy khách thứ nhất đến một trong số các nút máy chủ (“nút thứ nhất”), dữ liệu trạng thái hiển thị biểu diễn hiển thị được chia sẻ như được hiển thị bằng máy khách thứ nhất; gửi dữ liệu trạng thái hiển thị từ nút thứ nhất đến máy khách thứ hai thông qua nút khác trong số các nút máy chủ (“nút thứ hai”); cập nhật màn hình của máy khách thứ hai bằng cách sử dụng dữ liệu trạng thái hiển thị để thể hiện hiển thị được chia sẻ; trả lời sự thay đổi về hiển thị được chia sẻ ở máy khách thứ hai, gửi dữ liệu trạng thái hiển thị được cập nhật từ máy khách thứ hai đến nút thứ hai, trong đó trạng thái hiển thị được cập nhật biểu diễn hiển thị được chia sẻ như được hiển thị bằng máy khách thứ hai; gửi trạng thái hiển thị được cập nhật từ nút thứ hai đến máy khách thứ nhất thông qua nút thứ nhất; và cập nhật màn hình của máy khách thứ nhất để thể hiện hiển thị được chia sẻ bằng cách sử dụng trạng thái hiển thị được cập nhật. Theo một khía cạnh, không có nút máy chủ nào là máy chủ cổng kết nối được tập trung; theo khía cạnh thay thế, ít nhất một trong số các nút là máy chủ cổng kết nối được tập trung.

Nút thứ nhất và thứ hai và ít nhất một trong nhiều nút máy chủ khác có thể bao gồm cụm máy chủ, nút thứ nhất và thứ hai có thể bao gồm ít nhất một phần của nhóm nút máy chủ trong cụm mà nút thứ nhất có thể gửi dữ liệu trạng thái hiển thị theo cách theo thứ tự tổng thể đến tất cả các nút máy chủ khác trong nhóm, và gửi dữ liệu trạng thái hiển thị có thể bao gồm nút thứ nhất gửi dữ liệu đến tất cả các nút máy chủ khác trong nhóm.

Nút thứ nhất và thứ hai và ít nhất một trong nhiều nút máy chủ khác có thể bao gồm cụm máy chủ, nút thứ nhất và thứ hai có thể bao gồm ít nhất một phần của nhóm nút máy chủ trong cụm mà nút thứ hai có thể gửi trạng thái hiển thị được cập nhật theo cách theo thứ tự tổng thể đến tất cả các nút máy chủ khác trong nhóm, và gửi trạng

thái hiển thị được cập nhật có thể bao gồm nút thứ hai gửi trạng thái hiển thị được cập nhật đến tất cả các nút máy chủ khác trong nhóm.

Trước khi thể hiện hiển thị được chia sẻ trên màn hình của máy khách thứ hai, phương pháp có thể bao gồm gửi từ máy khách thứ nhất đến máy khách thứ hai thông qua nút thứ nhất và thứ hai thông báo rằng hiển thị chia sẻ như được hiển thị bằng máy khách thứ nhất sẵn sàng được chia sẻ với máy khách thứ hai.

Nút thứ nhất và thứ hai và ít nhất một trong nhiều nút máy chủ khác có thể bao gồm cụm máy chủ, nút thứ nhất và thứ hai có thể bao gồm ít nhất một phần của nhóm nút máy chủ trong cụm mà nút thứ nhất có thể gửi thông báo theo cách theo thứ tự tổng thể đến tất cả các nút máy chủ khác trong nhóm, và gửi thông báo có thể bao gồm nút thứ nhất gửi thông báo đến tất cả các nút máy chủ khác trong nhóm.

Trước khi nút thứ nhất gửi dữ liệu trạng thái máy khách thứ hai thông qua nút thứ hai, phương pháp này có thể bao gồm việc truy cập, sử dụng nút thứ nhất, mã nhận dạng nút nhận dạng nút thứ hai, trong đó nút thứ nhất và thứ hai bao gồm ít nhất một phần của cụm máy chủ và trong đó mã nhận dạng nút bao gồm ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả và có thể truy cập bằng tất cả nút máy chủ trong cụm máy chủ.

Thông tin hội viên của cụm có thể bao gồm mã nhận dạng nút nhận dạng duy nhất một trong số các nút máy chủ trong cụm máy chủ; và mã nhận dạng cụm nhận dạng duy nhất cụm máy chủ mà các nút máy chủ thuộc về.

Mỗi nút máy chủ trong cụm máy chủ có thể lưu trữ theo cách lâu dài phiên bản riêng của nó về thông tin hội viên của cụm một cách cục bộ.

Theo khía cạnh khác, sáng chế đề xuất hệ thống bảo mật vật lý, bao gồm máy khách thứ nhất có màn hình; máy khách thứ hai có màn hình; và nhiều nút máy chủ, trong đó một trong số các nút máy chủ (“nút thứ nhất”) giao tiếp với màn hình thứ nhất và một nút khác trong số các nút máy chủ (“nút thứ hai”) giao tiếp với màn hình thứ hai, trong đó máy khách thứ nhất và thứ hai và nút thứ nhất và thứ hai được cấu hình để gửi, từ máy khách thứ nhất đến nút thứ nhất, dữ liệu trạng thái hiển thị biểu diễn hiển thị được chia sẻ như được hiển thị trên màn hình của máy khách thứ nhất; gửi dữ

liệu trạng thái hiển thị từ nút thứ nhất đến máy khách thứ hai thông qua nút thứ hai; cập nhật màn hình của máy khách thứ hai bằng cách sử dụng dữ liệu trạng thái hiển thị để thể hiện hiển thị được chia sẻ; trả lời sự thay đổi theo hiển thị được chia sẻ ở máy khách thứ hai, gửi dữ liệu trạng thái hiển thị được cập nhật từ máy khách thứ hai đến nút thứ hai, trong đó trạng thái hiển thị được cập nhật biểu diễn hiển thị được chia sẻ như được hiển thị trên màn hình của máy khách thứ hai; gửi trạng thái hiển thị được cập nhật từ nút thứ hai đến máy khách thứ nhất thông qua nút thứ nhất; và cập nhật màn hình của máy khách thứ nhất để thể hiện hiển thị được chia sẻ bằng cách sử dụng trạng thái hiển thị được cập nhật. Theo một khía cạnh, không có nút máy chủ nào là máy chủ cổng kết nối được tập trung; theo khía cạnh thay thế, ít nhất một trong số các nút máy chủ là máy chủ cổng kết nối.

Theo khía cạnh khác, sáng chế đề xuất vật ghi có thể đọc được bằng máy tính không tạm thời mã hóa trên đó các câu lệnh và các chỉ dẫn để làm cho bộ xử lý thực hiện phương pháp chia sẻ hiển thị (“hiển thị được chia sẻ”) sử dụng hệ thống bảo mật vật lý mà bao gồm nhiều nút máy chủ, phương pháp bao gồm việc gửi, từ máy khách thứ nhất đến một trong số các nút máy chủ (“nút thứ nhất”), dữ liệu trạng thái hiển thị biểu diễn hiển thị được chia sẻ như được hiển thị bằng máy khách thứ nhất; gửi dữ liệu trạng thái hiển thị từ nút thứ nhất đến máy khách thứ hai qua nút khác trong số các nút máy chủ (“nút thứ hai”); cập nhật màn hình của máy khách thứ hai sử dụng dữ liệu trạng thái hiển thị để thể hiện hiển thị được chia sẻ; trả lời lại sự thay đổi về hiển thị được chia sẻ ở máy khách thứ hai, gửi dữ liệu trạng thái hiển thị được cập nhật từ máy khách thứ hai đến nút thứ hai, trong đó trạng thái hiển thị được cập nhật biểu diễn hiển thị được chia sẻ như được hiển thị bằng máy khách thứ hai; gửi trạng thái hiển thị được cập nhật từ nút thứ hai đến máy khách thứ nhất qua nút thứ nhất; và cập nhật màn hình của máy khách thứ nhất để thể hiện hiển thị được chia sẻ sử dụng trạng thái hiển thị được cập nhật.

Phần bản chất kỹ thuật này không nhất thiết mô tả toàn bộ phạm vi của tất cả các khía cạnh. Các khía cạnh, dấu hiệu và các ưu điểm khác sẽ rõ ràng đối với các chuyên gia trong lĩnh vực này dựa trên việc xem xét phần mô tả sau đây về các phương án cụ thể.

Mô tả vắn tắt các hình vẽ

Trong các hình vẽ kèm theo minh họa một hoặc nhiều phương án lấy làm ví dụ:

Fig.1 là sơ đồ khối về hệ thống bảo mật vật lý phân tán theo một phương án.

Fig.2 là sơ đồ khối về giao thức thích hợp được sử dụng bởi hệ thống trên Fig.1.

Fig.3 là sơ đồ theo thứ tự UML thể hiện làm thế nào hệ thống trên Fig.1 chia sẻ các cài đặt giữa các người dùng của hệ thống khác nhau.

Fig.4 là sơ đồ theo thứ tự UML thể hiện làm thế nào hệ thống trên Fig.1 chia sẻ trạng thái giữa các người dùng của hệ thống khác nhau.

Fig.5 là sơ đồ theo thứ tự UML thể hiện làm thế nào hệ thống trên Fig.1 chia sẻ hiển thị giữa các người dùng của hệ thống khác nhau.

Fig.6 là sơ đồ theo thứ tự UML thể hiện làm thế nào hệ thống trên Fig.1 chia sẻ các dòng giữa các người dùng của hệ thống khác nhau.

Fig.7 là hiển thị được quan sát bởi người dùng của hệ thống trên Fig.1.

Fig.8 là phương pháp chia sẻ dữ liệu trong hệ thống bảo mật vật lý theo phương án khác.

Fig.9 là phương pháp tham gia lại tự động vào cụm, theo phương án khác.

Fig.10 là sơ đồ theo thứ tự UML thể hiện làm thế nào hệ thống trên Fig.1 chia sẻ hiển thị không được giám sát với người dùng của hệ thống.

Fig.11 là phương pháp tương tác với màn hình không được giám sát trong hệ thống bảo mật vật lý mà bao gồm nhiều nút máy chủ, theo phương án khác.

Fig.12 là phương pháp chia sẻ sự hiển thị sử dụng hệ thống bảo mật vật lý bao gồm nhiều nút máy chủ, theo phương án khác.

Mô tả chi tiết sáng chế

Các thuật ngữ về hướng như “đỉnh”, “đáy”, “phía trên”, “phía dưới”, “thẳng đứng”, và “phía bên” được sử dụng trong phần mô tả sau đây với mục đích chỉ mang tính tham khảo tương đối, và không nhằm đề xuất sự giới hạn bất kỳ làm thế nào vật dụng bất kỳ được đặt trong quá trình sử dụng, hoặc được lắp trong tổ hợp lắp ráp hoặc

tương đối với môi trường. Ngoài ra, thuật ngữ “ghép” và các biến thể của nó như “được ghép”, “các cặp”, và “ghép” như được sử dụng trong phần mô tả này nhằm bao gồm các kết nối trực tiếp và gián tiếp trừ phi có quy định khác. Ví dụ, nếu thiết bị thứ nhất được ghép với thiết bị thứ hai, mà việc ghép có thể thông qua kết nối trực tiếp hoặc thông qua kết nối gián tiếp qua các thiết bị và kết nối khác. Tương tự, nếu thiết bị thứ nhất được ghép giao tiếp với thiết bị thứ hai, việc giao tiếp có thể thông qua kết nối trực tiếp hoặc thông qua kết nối gián tiếp qua các thiết bị và kết nối khác.

Ngay khi hệ thống giám sát phát triển bao gồm một số máy quay nhất định, nó trở nên phi thực tế hoặc không thể vận hành hệ thống giám sát bằng cách sử dụng một máy chủ do khả năng lưu trữ và các giới hạn công suất xử lý. Do đó, để cung cấp số máy quay tăng, các máy chủ bổ sung được bổ sung vào hệ thống. Việc này dẫn đến một số vấn đề.

Ví dụ, người dùng của hệ thống giám sát có thể muốn xem người khác xem gì (người dùng “xem”) và video dòng được chụp bằng cách sử dụng máy quay hoặc được lưu trữ trên máy chủ thậm chí người dùng không được lần lượt kết nối trực tiếp với máy quay hoặc máy chủ đó. Tương tự, người dùng có thể muốn có thể truy cập các trạng thái người dùng (ví dụ: liệu người dùng khác của hệ thống đăng nhập vào hệ thống) và các sự kiện của hệ thống (ví dụ: liệu liệu tiếng động có được tạo ra) mà xảy ra ở bất kỳ đâu trong hệ thống, thậm chí chúng bắt đầu ở máy chủ mà người dùng không được kết nối trực tiếp đến. Trong hệ thống giám sát thông thường được nhân ra bằng cách bổ sung nhiều máy chủ, cách thông thường để tạo ra chức năng này là thêm máy chủ cổng kết nối được tập trung vào hệ thống. Máy chủ cổng kết nối được tập trung gửi các sự kiện của hệ thống, trạng thái người dùng, các hiển thị, và video từ một máy chủ trong hệ thống đến máy chủ khác thông qua chính nó, theo đó cho phép người dùng truy cập hoặc xem các sự kiện, trạng thái, hiển thị và video này bất chấp máy chủ cụ thể nào mà người dùng kết nối trực tiếp đến. Tuy nhiên, việc sử dụng máy chủ cổng kết nối được tập trung tạo ra hệ thống giám sát một điểm hỏng, do nếu máy chủ cổng kết nối được tập trung hỏng thì sau đó các sự kiện, trạng thái, hiển thị và video không thể chia sẻ được. Việc sử dụng máy chủ cổng kết nối được tập trung cũng làm tăng giá thành của hệ thống giám sát, do máy chủ được bổ sung vào hệ thống và được dành cho để tạo ra chức năng của máy chủ cổng kết nối được tập trung.

Người dùng cũng có thể muốn các cài đặt chung (ví dụ: thông tin truy cập người dùng ở dạng tên người dùng, mật khẩu, quyền truy cập, v.v) được đồng bộ hóa qua nhiều máy chủ trong hệ thống. Trong hệ thống giám sát thông thường được nhân rộng bằng cách bổ sung nhiều máy chủ hơn, chức năng này được tạo ra bằng cách thay đổi các cài đặt thủ công từ một máy chủ sang các máy chủ khác, hoặc bằng cách sử dụng máy chủ giám sát được tập trung mà lưu trữ tất cả các cài đặt này mà các máy chủ khác giao tiếp nếu cần để tìm kiếm các cài đặt này. Các cài đặt thay đổi thủ công có vấn đề do độ trễ đồng bộ hóa tương đối rộng, độ khó cài đặt và sử dụng, và do độ trễ đồng bộ hóa lớn làm yếu sự dư thừa của hệ thống. Việc sử dụng máy chủ giám sát được tập trung gặp các vấn đề giống như việc sử dụng máy chủ cổng kết nối được tập trung, như được thảo luận trên đây.

Một số phương án được mô tả ở đây được đề xuất ở hệ thống bảo mật vật lý phân tán, như hệ thống giám sát, mà có thể tự động chia sẻ dữ liệu như các hiển thị, video, các sự kiện của hệ thống, trạng thái người dùng, và các cài đặt người dùng giữa hai hoặc nhiều nút máy chủ trong hệ thống mà không phụ thuộc vào máy chủ được tập trung như các máy chủ giám sát hoặc máy chủ giám sát được thảo luận trên đây. Các phương án này được đề xuất ở hệ thống giám sát ngang hàng mà người dùng kết nối qua các máy khách đến các nút máy chủ, như các thiết bị ghi video mạng, máy quay và máy chủ. Các nút máy chủ được nhóm với nhau trong cụm, với mỗi nút máy chủ trong cụm có thể chia sẻ dữ liệu với các nút máy chủ khác trong cụm. Để chia sẻ dữ liệu này, một trong số các nút máy chủ chạy các dịch vụ trao đổi dữ liệu dựa trên dây giao thức chia sẻ dữ liệu giữa các nút máy chủ theo các cách khác nhau tùy thuộc vào liệu dữ liệu biểu diễn các hiển thị, video, các sự kiện của hệ thống, trạng thái người dùng, hoặc các cài đặt người dùng. Các hình vẽ từ Fig.1 đến Fig.10 mô tả các phương án này.

Theo các phương án thay thế, một số công nghệ được sử dụng để chia sẻ các hiển thị giữa các nút máy chủ khác nhau có thể áp dụng với các mạng trung ương (nghĩa là, các mạng bao gồm máy chủ trung tâm) và các mạng ngang hàng như các mạng được thể hiện trên các hình vẽ từ Fig.1 đến Fig.9. Fig.10 và Fig.11 mô tả các phương án này.

Tham chiếu đến Fig.1 thể hiện hệ thống bảo mật vật lý phân tán ở dạng hệ thống giám sát 100, theo một phương án. Hệ thống 100 bao gồm ba máy khách 102a-c (máy khách thứ nhất 102a đến máy khách thứ ba 102c và “các máy khác 102” chung), sáu máy chủ 104a-f (máy chủ thứ nhất 104a đến máy chủ thứ sáu 104f và “các máy chủ 104” chung), ba máy quay nút máy chủ 106a-c (máy quay nút thứ nhất 106a đến máy quay nút thứ ba 106c và “các máy quay nút 106” chung), và năm máy quay không nút 114.

Một trong số các máy quay nút 106 và máy chủ 104 bao gồm bộ xử lý 110 và bộ nhớ 112 mà được nối theo cách giao tiếp với nhau, với bộ nhớ 112 mã hóa trên đó các câu lệnh và các chỉ dẫn để làm cho bộ xử lý 110 thực hiện các phương án các phương pháp được mô tả ở đây. Các máy chủ 104 và các máy quay nút 106 được nhóm thành ba cụm 108a-c (“các cụm 108” chung): các máy chủ từ một đến ba 104a-c được nối theo cách giao tiếp với nhau để tạo thành cụm thứ nhất 108a; các máy chủ thứ tư đến thứ sáu 104d-f được nối theo cách giao tiếp với nhau để tạo thành cụm thứ hai 108b; và ba máy quay nút 106 được nối theo cách giao tiếp với nhau để tạo thành cụm thứ ba 108c. Các máy chủ thứ nhất đến thứ ba 104a-c được tham chiếu đến là “các thành phần” của cụm thứ nhất 108a; các máy chủ thứ tư đến thứ sáu 104d-f được tham chiếu đến là “các thành phần” của cụm thứ hai 108b; và các máy quay nút từ thứ nhất đến thứ ba 106a-c được tham chiếu đến là “các thành phần” của cụm thứ ba 108c.

Một trong số các máy chủ 104 và các máy quay nút 106 là “nút máy chủ” trong đó mỗi máy nhận biết sự có mặt của các thành phần khác của cụm của nó 108 và có thể gửi dữ liệu đến các thành phần khác của cụm của nó 108; ngược lại, các máy quay không nút 114 không là các nút máy chủ mà trong đó chúng chỉ nhận biết các máy chủ 104a,b,c,d,f mà chúng được kết nối trực tiếp. Theo phương án được mô tả, các nút máy chủ nhận biết tất cả các thành phần khác của cụm 108 do có truy cập đến thông tin hội viên của cụm, mà liệt kê tất cả các nút máy chủ trong cụm 108. Thông tin hội viên của cụm được lưu trữ theo cách lâu dài hoặc cục bộ trên mỗi nút máy chủ, mà cho phép mỗi nút máy chủ tham gia lại một cách tự động vào cụm của nó 108 nếu nó khởi động lại trong quá trình hoạt động của hệ thống 100. Tham chiếu sau đây đến “nút” là viện dẫn đến “nút máy chủ” trừ khi có quy định khác.

Trong khi theo phương án được mô tả, không có thành phần nào trong các cụm 108 tham gia vào giao tiếp liên cụm, theo các phương án thay thế (không được thể hiện) các thành phần của các cụm khác nhau 108 có thể chia sẻ dữ liệu với nhau. Theo phương án được mô tả, các máy chủ 104 là các máy chủ thương mại có sẵn và các máy quay 106,114 được sản xuất bởi Avigilon™ Corporation, Vancouver, Canada; tuy nhiên, theo các phương án thay thế, các loại máy chủ thích hợp khác 108 và các máy quay 106,114 có thể được sử dụng.

Máy khách thứ nhất 102a được nối theo cách giao tiếp với các cụm thứ nhất và thứ hai 108a,b do được ghép giao tiếp với các máy chủ thứ nhất và thứ tư 104a,d, mà là các thành phần của các cụm đó 108a,b; máy khách thứ hai 102b được ghép giao tiếp với tất cả ba cụm 108 do được nối theo cách giao tiếp với các máy chủ thứ hai và thứ tư 104b,d và máy quay nút thứ nhất 106a, mà là các thành phần của các cụm đó 108; và máy khách thứ ba 102c được ghép giao tiếp với cụm thứ hai và thứ ba 108b,c do được nối theo cách giao tiếp với máy chủ thứ năm 104e và máy quay nút thứ hai 106b, là các phần của các cụm đó 108b,c. Như được thảo luận chi tiết hơn sau đây, trong một trong số các cụm bất kỳ được đưa ra 108a-c, một trong số các nút này chạy các dịch vụ cho phép các nút giao tiếp với nhau theo sự thích hợp về giao thức 200 (được thể hiện trên Fig.2), mà cho phép một nút bất kỳ chia sẻ dữ liệu, liệu dữ liệu có là các hiển thị, video, các sự kiện của hệ thống, các trạng thái người dùng, các cài đặt người dùng, hoặc loại dữ liệu khác, với nút khác bất kỳ sử dụng việc tính toán phân tán; nghĩa là, không sử dụng máy chủ được tập trung. Mỗi trong số các nút có truy cập vào thông tin hội viên của cụm mà nhận biết tất cả các nút tạo thành một phần của cùng cụm 108; bằng cách truy cập thông tin hội viên của cụm này, dữ liệu có thể được chia sẻ và đồng bộ hóa giữa tất cả các nút của cụm 108.

Fig.2 thể hiện dây giao thức 200 được sử dụng bởi các nút của hệ thống 100. Dây giao thức 200 được chia thành ba lớp và bao gồm các giao thức sau đây, như được tóm tắt trong Bảng 1:

Bảng 1: Tóm tắt dây giao thức 200

<u>Tên giao thức</u>	<u>Tầng giao thức</u>	<u>Nhận dữ liệu từ các giao thức và ứng dụng này</u>	<u>Gửi dữ liệu đến các giao thức này</u>
UDP 202	Giao vận	Giao thức khám phá 206, Giao thức nút 210, Giao thức đồng bộ 214	N/A
TCP/HTTP 204	Giao vận	Giao thức nút 210, Giao thức tán gẫu 208, Giao thức hội viên 212, Giao thức nhất quán 216, Giao thức trạng thái 218	N/A
Giao thức khám phá 206	Hỗ trợ cụm	Giao thức nút 210	UDP 202
Giao thức tán gẫu 208	Hỗ trợ cụm	Giao thức hội viên 212, Giao thức nhất quán 216, Giao thức trạng thái 218	TCP/HTTP 204, Giao thức nút 210, Giao thức hội viên 212
Giao thức nút 210	Hỗ trợ cụm	Ứng dụng dòng cụm 220, Đồng bộ 214, Giao thức nhất quán 216, Giao thức hội viên 212, Giao thức trạng thái 218, Giao thức tán gẫu 208	UDP 202, TCP/HTTP 204, Giao thức khám phá 206

Giao thức hội viên 212	Hỗ trợ cụm	Giao thức đồng bộ 214, Giao thức tán gẫu 208, Giao thức trạng thái 218, Giao thức nhất quán 216	Giao thức tán ngẫu 208, Giao thức nút 210, TCP/HTTP 204
Giao thức đồng bộ 214	Đồng bộ dữ liệu	Hiện thị được chia sẻ và ứng dụng hợp tác 222, Các sự kiện được chia sẻ và ứng dụng cảnh báo 224, Ứng dụng chia sẻ hiển thị không được giám sát 225	UDP 202, Giao thức nút 210, Giao thức hội viên 212
Giao thức nhất quán 216	Đồng bộ dữ liệu	Ứng dụng cài đặt được chia sẻ 226, Ứng dụng đối tượng người dùng được chia sẻ 228, Ứng dụng chia sẻ hiển thị không được giám sát 225	Giao thức nút 210, Giao thức hội viên 212, Giao thức tán ngẫu 208, TCP/HTTP 204
Giao thức trạng thái 218	Đồng bộ dữ liệu	Ứng dụng hệ thống (thiết bị, máy chủ, v.v.) 230	Giao thức tán ngẫu 208, Giao thức hội viên 212, Giao thức nút 210, TCP/HTTP 204

Phần mô tả về chức năng và hoạt động của mỗi giao thức trong dãy giao thức 200 như sau.

Tầng giao vận

Tầng giao vận tương ứng với tầng thứ tư của mô hình kết nối các hệ thống mở (Open Systems Interconnection - OSI), và đóng vai trò cung cấp các dịch vụ truyền dữ liệu tin cậy giữa các nút đến các tầng hỗ trợ cụm, đồng bộ hóa dữ liệu, và ứng dụng. Tầng giao vận trong hệ thống 100 bao gồm các giao thức UDP 202 và TCPIHTTP 204.

Tầng hỗ trợ cụm

Tầng hỗ trợ cụm bao gồm các giao thức được sử dụng để phát hiện các nút, xác nhận sự tồn tại, kiểm tra tính linh động của nút, xác nhận rằng liệu nút có là thành phần của một trong số các cụm 108, và xác định làm thế nào truyền dữ liệu giữa các nút.

Giao thức khám phá 206

Giao thức khám phá 206 dựa trên phiên bản 1.1 của WS-Giao thức khám phá được công bố bởi Tổ chức tiên tiến của các tiêu chuẩn thông tin cấu trúc (OASIS), toàn bộ được kết hợp ở đây bằng cách viện dẫn. Theo phương án được mô tả, định dạng XML được sử dụng trong tiêu chuẩn được công bố mà được thay thế bằng mã hóa Google™ Protobuf.

Giao thức khám phá 206 cho phép nút bất kỳ hệ thống 100 nhận dạng các nút khác trong hệ thống 100 bằng cách truyền đa điểm các tin nhắn Probe đến các nút khác đó và đợi chúng trả lời. Theo cách khác, nút có thể được truyền rộng điểm tin nhắn Hello khi tham gia vào hệ thống 100 để cảnh báo các nút khác sự có mặt của nó mà không yêu cầu các nút đó để trước hết truyền đa điểm tin nhắn Probe. Cả hai tin nhắn Probe và Hello được mô hình hóa trên WS-Giao thức khám phá được công bố bởi OASIS.

Giao thức tán gẫu 208

Giao thức tán gẫu 208 là giao thức truyền mà truyền dữ liệu từ một trong số các nút đến tất cả các nút của cụm 108 bằng cách thực hiện trao đổi dữ liệu ngẫu nhiên giữa các cặp nút trong cụm 108. Giao thức tán gẫu 208 giao tiếp linh động bằng cách trao đổi dữ liệu “trạng thái nhịp” ở dạng dữ liệu đếm nhịp với mỗi nút, mà cho phép các nút xác định khi một trong số các nút trong cụm 108 bỏ bất ngờ (ví dụ: do rơi máy

chủ). Giao thức tán gẫu 208 cũng giao tiếp dữ liệu “trạng thái ứng dụng” như thông tin băm mức đỉnh được sử dụng bằng Giao thức nhất quán 216 và các mã nhận dạng thực thể trạng thái và các số phiên bản của chúng được sử dụng bằng Giao thức trạng thái 218 để xác định khi nào đồng bộ hóa dữ liệu giữa các nút, như được thảo luận chi tiết hơn sau đây. Sự lan truyền dữ liệu bằng cách sử dụng Giao thức tán gẫu 208 cuối cùng lan truyền đến tất cả các nút trong cụm 108 thông qua các trao đổi nút với nút định kỳ.

Sự trao đổi dữ liệu giữa hai nút bất kỳ của cụm 108 bằng cách sử dụng Giao thức tán gẫu 208 bao gồm việc thực hiện hai cuộc gọi thủ tục từ xa (RPC) từ nút thứ nhất (“Nút A”) đến nút thứ hai (“Nút B”) trong cùng cụm 108, như sau:

1. Nút A gửi tin nhắn GreetingReq đến Nút B, mà chứa danh sách các tài liệu với tất cả các nút trong cụm 108 mà Nút A nhận biết. Với mỗi nút, tài liệu bao gồm mã nhận dạng nút duy nhất và thông tin phiên bản được gia tăng mỗi lần trạng thái nhịp hoặc trạng thái ứng dụng với nút đó thay đổi. Thông tin phiên bản có thể là, ví dụ, số phiên bản một chiều hoặc vector phiên bản đa chiều. Việc sử dụng vector phiên bản cho phép tài liệu tóm tắt lược sử dụng các thay đổi trạng thái mà nút trải qua.

2. Nút B gửi tin nhắn GreetingRsp đến Nút A, bao gồm:

- (a) danh sách tài liệu với các nút mà Nút B muốn nhận nhiều thông tin từ Nút A, mà Nút B xác định từ thông tin phiên bản được gửi đến nó trong tin nhắn GreetingReq;

- (b) danh sách tài liệu với các nút mà Nút A không biết về phần dạng của cụm 108;

- (c) danh sách một hoặc hai trạng thái nhịp và ứng dụng mà sẽ mang Nút A cập nhật mà nó hết hạn thông tin; và

- (d) danh sách các nút mà Nút A tin phần tạo thành của cụm 108 trừ khi Nút B biết đã bị loại bỏ khỏi cụm 108.

3. Tiếp đó Nút A gửi tin nhắn ClosureReq đến Nút B, trong đó Nút A gửi:

- (a) danh sách các tài liệu cho các nút mà Nút A muốn nhận thông tin từ Nút B (ví dụ, Nút A có thể yêu cầu thông tin cho các nút mà Nút A không nhận biết được cho đến khi Nút B đã Nút A tin nhắn GreetingRsp);

(b) danh sách các trạng thái sẽ mang Nút B cập nhật trên các nút mà nó hết hạn thông tin; và

(c) danh sách các nút mà Nút B tin rằng phần tạo thành của cụm 108 trừ khi Nút A biết rằng đã bị loại khỏi cụm 108.

4. Sau đó Nút B gửi tin nhắn ClosureRsp đến Nút A, trong đó Nút B gửi:

(a) danh sách các trạng thái sẽ mang Nút A cập nhật trên các nút mà nó hết hạn trên đó, trả lời lại yêu cầu của Nút A trong ClosureReq; và

(b) danh sách các nút mà bị loại bỏ khỏi cụm 108 do GreetingRsp.

Sau khi các nút A và B trao đổi các RPC, chúng sẽ có các danh sách nút hoạt động giống nhau mà bao gồm các phiên bản mới nhất của trạng thái nhịp và trạng thái ứng dụng với tất cả các nút trong cụm 108 mà cả hai đã biết về trước khi các RPC và không bị loại bỏ khỏi cụm 108.

Giao thức nút 210

Giao thức nút 210 chịu trách nhiệm tạo sự hiển thị của hình học topo mạng của hệ thống 100 cho mỗi nút, mà cung cấp cho mỗi nút ánh xạ mạng cho phép nó giao tiếp với nút khác bất kỳ trong hệ thống 100. Theo một số phương án, bản đồ mạng bằng chỉ đường đi. Bản đồ mạng tham chiếu các điểm cuối giao tiếp, là địa chỉ (IP/FQDN), số cổng, và giao thức mà nút có thể đạt qua mạng IP kết nối các nút.

Giao thức nút 210 xử lý việc này theo ba cách:

1. thông qua “Trao đổi Poke”, như được mô tả chi tiết hơn sau đây;

2. thông qua giao thức khám phá 206, mà thông báo giao thức nút 210 khi nút tham gia hoặc rời hệ thống 100. Khi nút tham gia vào hệ thống 100 “Trao đổi Poke” được thực hiện với nút đó; và

3. thủ công, trả lời lại đầu vào của người dùng.

Trao đổi Poke bao gồm thực hiện theo giai đoạn các RPC sau đây với mục đích tạo các bản đồ mạng cho các nút:

1. Yêu cầu Poke, trong đó Nút A gửi đến Nút B Nút A tự hiển thị các nút khác đã biết đến Nút A, như được hiển thị bởi Nút A, sau đó Nút B cập nhật bản đồ mạng của nó về thông tin này; và

2. Trả lời Poke, trong đó Nút B gửi đến Nút A Nút B tự hiển thị và danh sách các nút khác đã biết đến Nút B, như được hiển thị bởi Nút B, sau đó Nút A cập nhật bản đồ mạng của nó về thông tin này.

Các RPC được thực hiện qua giao thức TCP/HTTP 204.

Để làm giảm sử dụng băng thông, thông tin nút chỉ được trao đổi giữa các nút Node A và B nếu thông tin nút thay đổi do lần cuối cùng nó được trao đổi.

Trao đổi Poke được thực hiện sau khi giao thức khám phá 206 thông báo giao thức nút 210 rằng nút vừa tham gia hệ thống 100 do giao thức khám phá 206 quảng cáo các điểm cuối giao tiếp của nút, nhưng không đảm bảo rằng nút có thể đạt được bằng cách sử dụng các điểm cuối giao tiếp đó. Ví dụ, các điểm cuối có thể không thể sử dụng do tường lửa. Việc thực hiện trao đổi Poke trên nút được nhận biết bằng cách sử dụng giao thức khám phá 206 xác nhận liệu các điểm cuối giao tiếp, trong thực tế, có thể sử dụng được.

Giao thức nút 210 cũng có thể xác nhận liệu điểm cuối giao tiếp UDP được quảng cáo có thể đạt được; tuy nhiên, giao thức nút 210 theo phương án được mô tả không thực hiện trao đổi Poke qua giao thức UDP 202.

Với nút được đưa ra bất kỳ trong cụm 108, bản đồ mạng liên quan đến các mã nhận dạng nút nút đến các điểm điểm cuối giao tiếp cho một trong số các nút trong cùng cụm 108. Do đó, các giao thức khác trong chồng giao thức 200 mà giao tiếp với giao thức nút 210 có thể đưa các tin nhắn đến nút khác trong cụm 108 chỉ bằng cách sử dụng mã nhận dạng nút của nút đó.

Giao thức hội viên 212

Giao thức hội viên 212 chịu trách nhiệm đảm bảo mỗi nút của cụm 108 duy trì thông tin hội viên của cụm với tất cả các nút của cụm 108, và cho phép các nút tham gia và rời khỏi cụm 108 qua các RPC. Thông tin hội viên của cụm được chia sẻ giữa các nút của cụm 108 sử dụng Giao thức trạng thái 218. Mỗi nút trong cụm 108

duy trì phiên bản riêng của nó của thông tin hội viên của cụm và học từ Giao thức trạng thái 218 thông tin hội viên của cụm được giữ bởi các nút khác trong cụm 108. Như được thảo luận chi tiết hơn sau đây, các phiên bản của thông tin hội viên của cụm được giữ bởi hai nút khác nhau không thể khớp nhau do phiên bản của thông tin hội viên của cụm được lưu trữ trên nút And mà được cập nhật gần đây có thể chưa được đồng bộ hóa với các phiên bản khác của cụm 108.

Đối với mỗi nút, thông tin hội viên của cụm bao gồm:

1. Danh sách hội viên của tất cả các nút của cụm 108, trong đó mỗi nút được biểu diễn bởi:

(a) mã nhận dạng nút, là duy nhất trong số tất cả các nút trong hệ thống 100;

(b) trạng thái của nút, là một trong:

(i) Phát hiện: nút là thành phần của cụm 108 nhưng chưa được đồng bộ hóa bằng các thành phần khác của cụm 108 do được khởi động;

(ii) Tham gia: nút trong quá trình tham gia vào cụm 108;

(iii) Đồng bộ: nút trong quá trình đồng bộ hóa dữ liệu sử dụng các giao thức Đồng bộ, Nhất quán, và trạng thái 214,216,218 với cụm 108 mà nó vừa tham gia;

(iv) Hiệu lực: nút vừa hoàn thành việc đồng bộ hóa thông tin hội viên của cụm và là nút có hiệu lực của cụm 108; và

(v) Thời gian chờ: nút trở nên không phản ứng và không là thành phần hoạt động của cụm 108 (nút giữ thành phần của cụm 108 cho đến khi bị loại bỏ bởi người dùng);

(c) thẻ phiên;

(d) số phiên bản của thông tin hội viên của cụm khi nút được tham gia vào cụm 108; và

(e) số phiên bản của thông tin hội viên của cụm mà lần cuối nó được thay đổi.

2. Danh sách bia liệt kê tất cả các nút bị loại bỏ từ cụm 108, trong đó mỗi nút bị loại bỏ được biểu diễn bằng:

(a) mã nhận dạng nút của nút đó; và

(b) phiên bản của thông tin hội viên của cụm của nút khi nút bị loại bỏ.

Theo phương án được mô tả, nút luôn là thành phần của cụm 108 mà bao gồm ít nhất bản thân nó; cụm 108 của một nút được đề cập đến là “chỉ một cụm”. Hơn nữa, trong khi theo phương án được mô tả, thông tin hội viên bao gồm danh sách hội viên và danh sách bia như được mô tả trên đây, theo các phương án thay thế (không được mô tả), thông tin hội viên có thể bao gồm khác nhau; ví dụ, theo một phương án thay thế, thông tin hội viên thiếu danh sách bia, trong khi theo phương án khác, trạng thái của nút có thể được mô tả khác nhau hơn trên đây.

Khi Nút A muốn đóng vai trò là nút máy chủ mới và muốn tham gia vào cụm 108 mà bao gồm Nút B, thì nó giao tiếp với Nút B và xảy ra như sau:

1. Nút A gửi bí mật của cụm đến Nút B, mà theo phương án được mô tả là khóa mà Nút B yêu cầu trước khi cho nút khác tham gia vào cụm của nó 108. Một trong số các máy khách 102 cung cấp bí mật của cụm cho Nút A. Do Nút B điều khiển sự truy cập của Nút đến cụm 108, Nút B đóng vai trò là “nút điều khiển hội viên”.

2. Các nút A và B trao đổi thông tin hội viên của chúng. Các phiên bản của thông tin hội viên trên các nút A và B được cập nhật để bao gồm các mã nhận dạng nút của Nút A và tất cả các nút của cụm 108 mà Nút A tham gia.

3. Trạng thái của nút A được thay đổi thành “tham gia” do Nút A tham gia vào cụm.

4. Mỗi lần được tham gia, trạng thái của Nút A thay đổi thành “Đồng bộ” do dữ liệu được thay đổi giữa Nút A và cụm 108 mà nó vừa tham gia. Nút B cũng cập nhật phiên bản của thông tin hội viên được lưu trữ trên tất cả các nút khác của cụm 108 bằng cách sử dụng Giao thức trạng thái 218. Quy trình cập nhật các phiên bản của thông tin hội viên được lưu trữ trên Nút A và tất cả các thành phần của cụm 108 mà Nút A tham gia được đề cập đến là “đồng bộ hóa” các phiên của thông tin hội viên được lưu trữ trên tất cả các nút này.

5. Sau khi việc đồng bộ hóa kết thúc, trạng thái của Nút A thay đổi thành Hiệu lực.

Tầng đồng bộ dữ liệu

Tầng đồng bộ dữ liệu bao gồm các giao thức mà cho phép dữ liệu được gửi giữa các nút trong cụm với các đảm bảo thứ tự khác nhau và các trao đổi hiệu suất. Các giao thức trong Tầng đồng bộ dữ liệu sử dụng trực tiếp trong tầng giao vận và tầng hỗ trợ cụm.

Giao thức đồng bộ 214

Giao thức xử lý đồng thời 214 được sử dụng để gửi dữ liệu ở dạng các tin nhắn từ Nút A đến Nút B trong hệ thống 100 sao cho các tin nhắn đi đến Nút B theo thứ tự mà Nút A có thể điều khiển, như thứ tự mà Nút A gửi các tin nhắn. Các dịch vụ truyền dữ liệu sử dụng giao thức xử lý đồng thời 214 chạy trên các luồng dịch vụ I/O ưu tiên cao chuyên dụng.

Theo phương án được mô tả, giao thức xử lý đồng thời 214 dựa trên sự thực thi của sự đồng bộ ảo đã biết là giao thức Totem, như được mô tả trong Agarwal DA, Moser LE, Melliar-Smith PM, Budhia RK, “The Totem Multiple-Ring Ordering and Topology Maintenance Protocol”, ACM Transactions on Computer Systems, 1998, trang 93 – 132. Trong giao thức xử lý đồng thời 214, các nút được nhóm với nhau thành các nhóm mà sau đây được gọi trong phần mô tả là “các vòng đồng bộ”, và nút trên vòng đồng bộ bất kỳ có thể hoàn toàn gửi các tin nhắn theo thứ tự đến các nút khác trên cùng vòng. Giao thức xử lý đồng thời 214 biến đổi giao thức Totem như sau:

1. Giao thức xử lý đồng thời 214 sử dụng cả hai mã nhận dạng dịch vụ và mã nhận dạng vòng để nhận dạng vòng đồng bộ. Mã nhận dạng dịch vụ nhận dạng tất cả các đối tượng của vòng Đồng bộ đưa ra, trong khi đó mã nhận dạng vòng nhận dạng đối tượng cụ thể của vòng Đồng bộ đưa ra. Ví dụ, mỗi thời điểm mà nút tham gia hoặc rời khỏi vòng Đồng bộ thì mã nhận dạng vòng của vòng sẽ thay đổi, nhưng không phải mã nhận dạng dịch vụ của nó. Mã nhận dạng dịch vụ cho phép nút truyền đa điểm hoàn toàn các tin nhắn theo thứ tự đến nhóm các nút chia sẻ cùng mã nhận dạng dịch vụ (nghĩa là nhóm nút thuộc về cùng vòng Đồng bộ).

2. Trong giao thức Totem, trong một số trường hợp khi các nút không gửi các tin nhắn mà vòng Đồng bộ được quan sát bởi các nút không phản ánh cấu hình vòng cuối cùng mà hội tụ khi các nút bắt đầu nhận tin. Giao thức xử lý đồng thời 214 cho phép các nút

gửi các tin nhắn dò cho nhau để làm cho các vòng Đồng bộ hội tụ trước khi gửi các tin nhắn không dò.

3. Giao thức Totem chỉ thể hiện các tin nhắn theo thứ tự được gửi đến tất cả các nút mà tạo thành phần vòng Đồng bộ. Ngược lại, giao thức xử lý đồng thời 214 sử dụng môđun gửi đi các tóm tắt tầng mạng từ giao thức xử lý đồng thời 214 bằng cách cung cấp giao diện để truyền rộng điểm đến tất cả các nút có thể đến trong hệ thống 100; truyền đơn hướng đến từng nút trong các nút bất kỳ trong hệ thống 100 bằng cách sử dụng danh sách các mã nhận dạng nút đến; và truyền một điểm đến một nút trong hệ thống 100 sử dụng mã nhận dạng nút của nó. Môđun gửi đi cũng hỗ trợ ghép kênh các dịch vụ trên cùng cổng IP sử dụng tin nhắn lọc và định tuyến bởi mã nhận dạng dịch vụ. Các tin nhắn ra từ nút được gửi đến tập hợp con của các nút có cùng mã dịch vụ trừ truyền đa điểm.

4. Giao thức xử lý đồng thời 214 sử dụng các tin nhắn phân mảnh và tải trọng người dùng lớn và kết tụ tạo ra các vấn đề gây ra do kích cỡ đơn vị truyền dẫn tối đa khoảng 1,500 byte.

5. Giao thức xử lý đồng thời 214 biến đổi các nút đường sử dụng các tin nhắn Tham gia, là các nút tin nhắn sử dụng trong giao thức Totem để tham gia vào vòng Đồng bộ:

(a) Các tin nhắn tham gia được gửi bằng các nút chỉ nếu chúng có mã nhận dạng nút thấp nhất trong tập hợp hiện thời của các nút hoạt động trong vòng Đồng bộ.

(b) Các nút không có mã nhận dạng nút thấp nhất trong các tin nhắn Tham gia đơn điểm hoạt động của chúng đến các nút có mã nhận dạng nút thấp nhất trong tập hợp hoạt động của chúng.

(c) Các tin nhắn tham gia bao gồm mã nhận dạng dịch vụ, và các nút không là phần của vòng Đồng bộ tương ứng mà không phản ứng.

So với giao thức Totem, các biến thể này giúp làm giảm băng thông tắc bởi các nút tham gia vòng Đồng bộ.

6. Giao thức xử lý đồng thời 214 phát hiện và các nút danh sách đen mà không thể tham gia vòng Đồng bộ do một số loại mất cấu hình. Ví dụ, nút có thể gửi đến, nhưng không nhận các tin nhắn từ, các nút khác có thể xuất hiện ở các nút khác để chỉ từng

gửi các tin nhắn dò do tất cả các tin nhắn khác, theo phương án, được thu hút, và theo đó sẽ bị liệt vào danh sách đen.

7. Giao thức xử lý đồng thời 214 thực hiện mã hóa tải trọng và xác thực các tin nhắn.

8. Giao thức xử lý đồng thời 214 giới hạn thời gian mà mỗi nút có thể giữ thẻ được sử dụng trong giao thức Totem; theo phương án được mô tả, mỗi nút có thể giữ thẻ trong 15ms.

9. Giao thức xử lý đồng thời 214 thực hiện giải thuật tránh tắc nghẽn thân thiện TCP.

Như được thảo luận chi tiết sau đây, hệ thống 100 sử dụng giao thức xử lý đồng thời để các hiển thị được chia sẻ và ứng dụng hợp tác 222 và các sự kiện được chia sẻ và ứng dụng cảnh báo 224; dữ liệu được chia sẻ giữa các thành phần của cụm 108 trong các ứng dụng này 222 không lâu dài và được chia sẻ có lợi nhanh và theo thứ tự đã biết.

Giao thức nhất quán 216

Giao thức nhất quán 216 được sử dụng để chia sẻ dữ liệu tự động và theo thời gian qua tất cả các nút của cụm 108 sao cho dữ liệu được chia sẻ sử dụng Giao thức nhất quán 216 cuối cùng được đồng bộ hóa trên tất cả các nút trong cụm 108. Các loại dữ liệu được chia sẻ sử dụng Giao thức nhất quán 216 được thảo luận chi tiết hơn sau đây trong các phần thảo luận ứng dụng cài đặt được chia sẻ 226 và ứng dụng các đối tượng người dùng được chia sẻ 228. Dữ liệu được chia sẻ bằng Giao thức nhất quán 216 được lưu trữ trong cơ sở dữ liệu trên mỗi nút, và mỗi mục nhập trong cơ sở dữ liệu bao gồm cặp trị số khóa mà chia khóa nhận dạng duy nhất trị số và các khóa độc lập với nhau. Giao thức nhất quán 216 đồng bộ hóa dữ liệu qua các nút trong khi giải quyết các biến đổi song song mà các nút khác nhau có thể thực hiện trên các cơ sở khác nhau. Như được thảo luận chi tiết hơn sau đây, giao thức nhất quán 216 đạt được điều này bằng cách trước hết được thông báo là các cơ sở dữ liệu không được đồng bộ hóa; thứ hai, tìm ra các mục nhập cơ sở dữ liệu cụ thể không được đồng bộ hóa; và thứ ba, tìm ra phiên bản mục nhập nào được đồng bộ hóa và được giữ lại gần nhất.

Để giải quyết các biến đổi song song mà xác định các thay đổi được tạo ra với các cơ sở dữ liệu, mỗi nút tham gia vào cụm 108 được chuyển cơ chế tạo phiên bản mang tính nhân quả được sử dụng để ghi lại khi nút tạo ra sự thay đổi với các dữ liệu và xác định

liệu các thay đổi có thể được tạo ra trước hoặc sau khi các thay đổi với cùng dữ liệu được tạo ra bằng các nút khác trong cụm 108. Theo phương án, một trong số các nút sử dụng đồng hồ cây thời gian (ITC) làm cơ chế tạo phiên bản mang tính nhân quả. Tuy nhiên, theo các phương án thay thế, các cơ chế tạo phiên bản khác như các đồng hồ vector và các vector phiên bản có thể được sử dụng. Hệ thống 100 cũng thực hiện đồng thời gian toàn cầu (UTC), mà được đồng bộ giữa các nút khác nhau sử dụng Giao thức thời gian mạng, để xác định thứ tự mà trong đó các thay đổi được tạo ra khi các ITC cho hai hoặc nhiều nút giống nhau. Các ITC được mô tả chi tiết hơn trong P. Almeida, C. Baquero, and V. Fonte, “Interval tree clocks: a logical clock for dynamic systems”, *Princi. Distri. Sys., Lecture Notes in Comp. Sci.*, vol. 5401, trang 259–274, 2008.

Thư mục mà giao thức nhất quán 216 đồng bộ hóa giữa các nút được chia thành các nhánh, một trong số chúng đượ đề cập đến là miền nhất quán sự kiện (ECD). Giao thức nhất quán 216 đồng bộ hóa một trong số các ECD độc lập với các ECD khác. Mỗi mục nhập cơ sở dữ liệu nằm trong ECD được đề cập đến là mục nhập nhất quán sự kiện (ECE). Mỗi ECE bao gồm khóa; thời điểm ghi dấu từ ITC và từ UTC, mà cả hai được cập nhật bất kỳ khi nào ECE được biến đổi; trị số băm của ECE tạo ra sử dụng, ví dụ, hàm Murmurhash; dữ liệu của nó; bia mà được thêm nếu và khi ECE bị xóa.

Trị số băm được sử dụng để so sánh các ECD và ECE tương ứng trên hai nút khác nhau để xác định liệu chúng có giống nhau không. Khi so sánh hai ECD tương ứng, các thông tin băm “mức trên” với các ECD đó được so sánh. Thông tin băm mức đỉnh với ECD trên nút đưa ra được tạo ra bằng cách băm tất cả các ECE trong ECD đó. Nếu các thông tin băm mức đỉnh phù hợp, thì các ECD giống nhau; nếu không, giao thức nhất quán 216 xác định các ECD khác nhau. Để xác định các ECE cụ thể trong các ECD khác nhau, các băm được có các khoảng giảm liên tục về các ECE trên cả hai nút. Khoảng thời gian mà các băm cuối cùng đủ co lại mà các ECE khác nhau giữa hai nút được tách và nhận biết. Danh sách bỏ qua hai hướng có thể được sử dụng, ví dụ, để xác định và so sánh các trị số băm của các khoảng thời gian ECD.

Hai nút giao tiếp sử dụng giao thức nhất quán 216 có thể sử dụng các RPC sau đây:

1. SetEntries: SetEntries truyền các ECE mới hoặc được cập nhật đến nút, mà chèn chúng vào các ECD thích hợp.
2. GetEntries: GetEntries truyền khóa hoặc khoảng của các khóa đến nút mà quay trở lại các, ECE tương ứng với một hoặc nhiều khóa đó.
3. SynEntries: SynEntries truyền khóa hoặc dải khoá đến nút, và hai nút tiếp đó so sánh các băm theo các dải giảm liên tục của các ECE để xác định các ECE nào khác nhau giữa hai nút, như được mô tả trên đây. Nếu các ECE khác nhau, các nút hợp nhất các ECE của chúng sao cho các ECE giống nhau được lưu trữ trên các nút bằng cách so sánh các dấu ấn thời gian ITC; nếu các dấu ấn thời gian ITC phù hợp, các nút so sánh các dấu ấn thời gian liên quan đến các ECE. Các dấu ấn thời gian này đóng vai trò là thông tin phiên bản cho phép hai nút chấp nhận các ECE mà làm biến đổi một cách gần nhất, như được chỉ ra bởi các thông tin phiên bản của ECE đó.

Khi nút thay đổi ECE, thì nút đó thường gọi SynEntries để thông báo cho các nút khác trong cụm 108 mà các ECE được thay đổi. Nếu một số cụm trong cụm 108 không có sẵn (ví dụ: chúng ngoại tuyến), thì giao thức tán gẫu 208 thay vì SynEntries được sử dụng để giao tiếp các thông tin băm mức đỉnh với các nút không có sẵn ngay khi chúng trở lại trực tuyến. Như nói đến trong phần này thảo luận về Giao thức tán gẫu 208 trong cụm 108 trên đây, một trong số các nút giữ thông tin băm mức đỉnh của nó, thì lan truyền sang các nút khác cùng với mã nhận dạng nút, thông tin phiên bản, và trạng thái co đập của tim bằng cách sử dụng Giao thức tán gẫu 208. Khi nút khác nhận băm này, nó so sánh thông tin băm mức đỉnh nhận được với thông tin băm mức đỉnh riêng của nó. Nếu các thông tin băm mức đỉnh giống nhau, các ECE trên cả hai nút phù hợp; nếu không, các ECE khác nhau.

Nếu ECE khác nhau, bắt chấp liệu việc này có xác định sử dụng SynEntries hay Giao thức tán gẫu 208, nút mà chạy SynEntries hoặc nhận thông tin băm mức đỉnh thì đồng nhất hóa các ECE.

Giao thức trạng thái 218

Như được thảo luận trên đây, Giao thức tán gẫu 208 chia sẻ thông qua cụm 108 các mã nhận dạng thực thể trạng thái và các số phiên bản của chúng (“cập thực thể

trạng thái”) với các nút trong cụm 108. Các mã nhận dạng thực thể minh họa có thể, ví dụ, biểu diễn các loại dữ liệu trạng thái khác nhau ở dạng các mục nhập trạng thái như lượng dự trữ bao nhiêu mà nút sẵn có; các thiết bị nào (như các máy quay không nút 114) được kết nối đến nút đó; các máy khách nào 102 được kết nối với nút đó; và thông tin hội viên của cụm. Khi một trong số các nút nhận dữ liệu này qua Giao thức tán gẫu 208, nó so sánh số phiên bản của cặp thực thể trạng thái với số phiên bản của mục nhập trạng thái tương ứng mà nó lưu trữ cục bộ. Nếu số phiên bản khác nhau, giao thức trạng thái 218 bắt đầu RPC (“Sync RPC”) với nút mà cặp thực thể trạng thái bắt đầu cập nhật mục nhập trạng thái.

Mục nhập trạng thái được đồng bộ hóa sử dụng giao thức trạng thái 218 được nhận biết duy nhất bằng cả hai con đường và mã nhận dạng nút. Không giống như dữ liệu được đồng nhất hóa bằng cách sử dụng giao thức nhất quán 216, nút mà mục nhập trạng thái mô tả chỉ là nút được phép biến đổi mục nhập trạng thái hoặc cặp thực thể trạng thái. Do đó, và không giống như các ECD và ECE được đồng nhất hóa sử dụng giao thức nhất quán 216, phiên bản của mục nhập trạng thái với Nút A được lưu trữ cục bộ trên Nút A luôn là phiên bản gần nhất của mục nhập trạng thái đó.

Nếu nút A biến đổi nhiều mục nhập trạng thái đồng thời, Giao thức trạng thái 218 đồng bộ hóa tất cả các mục nhập trạng thái cùng nhau đến Nút B khi Nút B gọi Sync RPC. Do đó, các mục nhập được thay đổi đồng thời có thể phụ thuộc vào nhau do chúng sẽ cùng được gửi đến Nút B để phân tích. Ngược lại, mỗi ECE được đồng bộ hóa bằng cách sử dụng Giao thức nhất quán 216 được đồng bộ hóa độc lập với các ECE khác, nên các ECE không thể phụ thuộc vào nhau do Nút B không thể phụ thuộc vào việc nhận các mục nhập theo thứ tự cụ thể.

Các ứng dụng

Mỗi nút trong hệ thống 100 chạy các dịch vụ mà thực hiện dãy giao thức 200 được mô tả trên đây. Trong khi theo phương án được mô tả, một dịch vụ được sử dụng cho một trong số các giao thức 202-218, theo các phương án thay thế (không được mô tả), các dịch vụ lớn hơn hoặc ít hơn có thể được sử dụng để thực hiện dãy giao thức 200. Một trong số các nút thực hiện bản thân dãy giao thức 200; do đó, hệ thống 100 được phân tán và ít bị tấn công dẫn đến hỏng một nút bất kỳ, mà ngược lại với các hệ

thống bảo mật vật lý thông thường sử dụng máy chủ được tập trung. Ví dụ, nếu một trong số các nút hỏng trong hệ thống 100 (“nút bị hỏng”), trên một trong số các nút còn lại, dịch vụ chạy Giao thức trạng thái 218 (“dịch vụ trạng thái”) sẽ xác định rằng nút bị hỏng ngoại tuyến bằng cách kiểm soát trạng thái co đập của tim của nút bị hỏng và sẽ giao tiếp việc hỏng này với dịch vụ chạy Nút và các Giao thức hội viên 210,212 trên một trong số các nút khác (“Dịch vụ nút” và “dịch vụ hội viên”, tương ứng). Các dịch vụ này trên mỗi nút thực hiện các Giao thức đồng bộ và nhất quán 214,216 (“dịch vụ đồng bộ” và “dịch vụ nhất quán”, tương ứng) sẽ sau đó dùng chia sẻ dữ liệu có nút bị hỏng cho đến khi nút bị hỏng quay trở lại trực tuyến và tham gia lại vào cụm của nó 108.

Sau đây mô tả các ứng dụng khác nhau 220-230 mà hệ thống 100 có thể thực hiện. Các ứng dụng 220-230 có thể được thực hiện làm các phương án khác nhau của phương pháp minh họa để chia sẻ dữ liệu 800 được mô tả trên Fig.8. Phương pháp 800 bắt đầu ở khối 802 và tiến hành phong tỏa 804 trong đó nút thứ nhất trong hệ thống 100 truy cập mã nhận dạng nút nhận dạng nút khác trong hệ thống 100. Cả hai nút thứ nhất và thứ hai đều là các thành phần của cùng cụm máy chủ 108. Các truy cập mã nhận dạng nút thứ nhất là một phần của thông tin hội viên của cụm mà nhận dạng tất cả các thành phần của cụm 108. Thông tin hội viên của cụm có thể truy cập bằng tất cả các thành phần của cụm 108. Theo phương án được mô tả, một trong số các thành phần của cụm 108 lưu trữ riêng phiên bản của nó về thông tin hội viên của cụm lâu dài hoặc cục bộ; tuy nhiên, theo các phương án thay thế (không được mô tả), thông tin hội viên của cụm có thể được lưu trữ một hoặc cả hai ở xa từ các nút và ở vị trí trung tâm. Sau khi truy cập mã nhận dạng nút cho nút thứ hai, nút thứ nhất gửi dữ liệu đến nút thứ hai ở khối 806, sau đó phương pháp 800 kết thúc ở khối 808. Ví dụ, khi sử dụng dịch vụ Nút được mô tả trên đây, các dịch vụ Đồng bộ và Nhất quán chạy trên nút thứ nhất có thể gửi dữ liệu đến nút thứ hai bằng cách sử dụng mã nhận dạng nút của nút thứ hai, và bằng cách ủy quyền cho dịch vụ nút chịu trách nhiệm kết hợp điểm cuối giao tiếp của nút thứ hai với mã nhận dạng nút của nó. Việc gửi dữ liệu từ nút thứ nhất đến nút thứ hai tại khối 806 có thể bao gồm một phần của việc trao đổi dữ liệu hai hướng, như khi dữ liệu được trao đổi theo Giao thức tán gẫu 208.

Ứng dụng cài đặt chia sẻ 226 và Ứng dụng đối tượng người dùng chia sẻ 228

Trong quá trình hoạt động của hệ thống 100, thông tin được lưu trữ theo cách lâu dài được truyền giữa các nút của cụm 108. Các ví dụ về thông tin thời gian thực này mà các ứng dụng cài đặt chia sẻ và đối tượng người dùng chia sẻ 226,228 chia sẻ giữa các nút là các cài đặt chia sẻ như các nguyên tắc thực hiện trả lời các sự kiện của hệ thống như gây cảnh báo và các đối tượng người dùng như tên người dùng, mật khẩu và chủ đề. Loại dữ liệu này (“Dữ liệu nhất quán”) được chia sẻ giữa các nút sử dụng Giao thức nhất quán 216; nói chung, dữ liệu nhất quán là dữ liệu không phải được chia sẻ theo thời gian thực hoặc theo tổng thứ tự, và được lưu trữ theo cách lâu dài bằng mỗi nút. Tuy nhiên, theo các phương án thay thế (không được mô tả), dữ liệu nhất quán có thể được lưu trữ không lâu dài.

Fig.3 thể hiện sơ đồ thứ tự UML 300 mà trong đó Dữ liệu nhất quán ở dạng cài đặt người dùng được chia sẻ giữa người dùng thứ nhất và thứ hai 302a,b (gọi chung là, “các người dùng 302”). Người dùng 302, máy khách thứ nhất và thứ hai 102a,b, và các máy chủ thứ nhất và thứ hai 104a,b, là nút thứ nhất và thứ hai trong ví dụ này, là các đối tượng trong sơ đồ 300. Các máy chủ 104a,b tạo thành một phần của cùng cụm 108a. Do các máy chủ 104a,b mà các máy khách 102a,b giao tiếp thì không được kết nối trực tiếp với nhau, Giao thức nhất quán 216 được sử dụng để truyền dữ liệu giữa hai máy chủ 104a,b, và do đó giữa hai người dùng 302. Mặc dù phương án được mô tả các cài đặt chia sẻ, theo phương án thay thế (không được mô tả) thì người dùng 302 có thể chia sẻ tương tự các đối tượng người dùng.

Sơ đồ 300 có hai khung 332a,b. Trong khung thứ nhất 332a, người dùng thứ nhất 302a ra lệnh cho máy khách thứ nhất 102a mở bảng cài đặt (tin nhắn 304), và máy khách 102a sau đó thực hiện thủ tục `SettingsOpenView()` (tin nhắn 306), mà truyền các cài đặt đến máy chủ thứ nhất 104a. Đồng thời, người dùng thứ hai 302b ra lệnh cho máy khách thứ hai 102b tương tự (các tin nhắn 308 và 310). Trong khung thứ hai 332b, các người dùng 302 biên tập đồng thời các cài đặt của chúng. Người dùng thứ nhất 302a biên tập các cài đặt bằng cho máy khách thứ nhất 102a chạy `UIEditSetting()` (tin nhắn 312), sau đó máy khách thứ nhất 102a cập nhật các cài đặt được lưu trữ trên máy chủ thứ nhất 104a bằng cách cho máy chủ thứ nhất 104a chạy `SettingsUpdateView()` (tin nhắn 314). Sau đó, máy chủ thứ nhất 104a chạy

ConsistencySetEntries() (tin nhắn 316), thực hiện thủ tục SetEntries và truyền các cài đặt được nhập bởi người dùng thứ nhất 302a vào máy chủ thứ hai 104b. Sau đó, máy chủ thứ hai 104b gửi các cài đặt được truyền vào máy khách thứ hai 102b bằng cách gọi SettingsNotifyViewUpdate() (tin nhắn 318), sau đó máy khách thứ hai 102b cập nhật người dùng thứ hai 302b (tin nhắn 320). Đồng thời, người dùng thứ hai 302b biến đổi tương tự các cài đặt và gửi các cài đặt đó đến máy chủ thứ nhất 104a sử dụng Giao thức nhất quán 216 (các tin nhắn 322, 324, 326, 328, và 330). Mỗi máy chủ 104a,b lưu trữ theo cách lâu dài các cài đặt người dùng sao cho chúng không phải được đồng bộ hóa lại giữa các máy chủ 104a,b nếu một trong hai máy chủ 104a,b khởi động lại.

Các sự kiện được chia sẻ và ứng dụng cảnh báo 224

Trong quá trình hoạt động của hệ thống 100, thông tin thời gian thực được tạo ra trong thời gian chạy được truyền giữa các nút của cụm 108. Các ví dụ về thông tin thời gian thực này mà các sự kiện được chia sẻ và ứng dụng cảnh báo 224 chia sẻ giữa các nút là trạng thái cảnh báo (nghĩa là liệu việc cảnh báo có bị xung đột ở đâu trong hệ thống 100); các sự kiện của hệ thống như sự chuyển động được phát hiện, liệu thiết bị (như một trong số các máy quay của nút 106) gửi dữ liệu số đến phần còn lại của hệ thống 100, liệu thiết bị (như bộ dò chuyển động) được kết nối với hệ thống 100, liệu thiết bị hiện có ghi lại, liệu sự cảnh báo cáo xảy ra hoặc được xác nhận bởi các người dùng 302 thực hiện kiểm soát trên hệ thống 100, liệu một trong số các máy chủ 104 có mắc lỗi, liệu thiết bị được kết nối với hệ thống mắc lỗi, liệu điểm giao dịch an toàn có xảy ra; và nút máy chủ đến các thông báo máy khách như liệu các cài đặt /dữ liệu có thay đổi, trạng thái ghi lại hiện tại, liệu dòng thời gian có được cập nhật, và truy vấn cơ sở dữ liệu có kết quả. Theo phương án, dữ liệu được truyền giữa các nút sử dụng giao thức xử lý đồng thời 214 được đề cập đến là “Dữ liệu đồng bộ”, được tạo ra tại thời gian chạy, và không được lưu lâu dài bởi các nút.

Fig.4 thể hiện sơ đồ thứ tự UML 400, trong đó thông báo cảnh báo giữa các máy chủ 104 sử dụng giao thức xử lý đồng thời 214. Các đối tượng trong sơ đồ này 400 là một trong số các máy quay không nút 114, ba máy chủ 104 trong cụm thứ nhất 108a, và máy khách thứ hai 102b, mà được kết nối với một trong số các máy chủ 104c trong cụm thứ nhất 108a.

Ở ba khung thứ nhất 402 của sơ đồ 400, một trong số các máy chủ 104 tham gia vào Vòng Đồng bộ có tên là “ServerState” sao cho trạng thái của bất kỳ trong các máy chủ 104 có thể được giap tiếp với bất kỳ trong các máy chủ khác 104; theo phương án được mô tả, trạng thái này sẽ được giao tiếp là “AlarmStateTriggered”, nghĩa là sự cảnh báo trên một trong số các máy chủ 108 gây ra do sự kiện mà máy quay không nút 114 vừa phát hiện. Tại khung 404, máy chủ thứ hai 104b chọn “chủ” cho ứng dụng cảnh báo; điều này có nghĩa là đó là máy chủ thứ hai 104b xác định liệu đầu vào từ máy quay không nút 114 thỏa mãn các điều kiện chuyển dịch đến trạng thái AlarmStateTriggered, và gửi đến các máy chủ khác 104a,c trong Vòng Đồng bộ tin nhắn cũng chuyển dịch chúng đến trạng thái AlarmStateTriggered.

Người dùng thứ hai 302b đăng nhập vào máy chủ thứ ba 104c sau khi các máy chủ 104 tham gia vào ServerState Vòng Đồng bộ (tin nhắn 406). Sau khi người dùng 302b đăng nhập vào, máy chủ thứ ba 104c tham gia vào Vòng Đồng bộ có tên là “ClientNotification” (tin nhắn 408); như được thảo luận chi tiết hơn sau đây, vòng này được sử dụng để giao tiếp các trạng thái hệ thống với người dùng 302b, trong khi đó Vòng Đồng bộ ServerState được sử dụng để giao tiếp chỉ giữa các máy chủ 104. Máy quay không nút 114 gửi đầu vào số, như chỉ dẫn mà cửa hoặc cửa sổ được mở, đến máy chủ thứ nhất 104a (tin nhắn 410), sau đó máy chủ thứ nhất 104a kiểm tra xem liệu đầu vào số này có thỏa mãn tập hợp các nguyên lý được sử dụng để xác định liệu có gây ra cảnh báo trong hệ thống 100 (tin nhắn 412). Theo phương án được mô tả, máy chủ thứ nhất 104a xác định cảnh báo nên được tạo ra, và theo đó gọi AlarmTrigger() (tin nhắn 414), mà cảnh báo máy chủ thứ hai 104b thay đổi các trạng thái. Sau đó, máy chủ thứ hai 104 chuyển dịch các trạng thái đến AlarmStateTriggered (tin nhắn 416) và gửi tin nhắn đến Vòng Đồng bộ ServerState mà ra lệnh cho hai máy chủ khác 104a,c để thay đổi trạng thái thành AlarmStateTriggered (khung 418). Sau khi ra lệnh cho các máy chủ khác 104a,c, máy chủ thứ hai 104b chạy AlarmTriggerNotification() (tin nhắn 420), làm cho máy chủ thứ hai 104b để cũng tham gia vào Vòng Đồng bộ ClientNotification (khung 422) và gửi tin nhắn đến Vòng Đồng bộ ClientState làm cho máy chủ thứ ba 104c, là máy chủ khác trên Vòng Đồng bộ ClientState, để chuyển dịch đến trạng thái “NotifyAlarmTriggered” (khung 424). Ngay khi máy chủ thứ ba 104c thay đổi thành trạng thái này thì nó thông báo trực tiếp cho máy khách thứ hai 102b

rằng cảnh báo vừa xảy ra, sắp xếp tin nhắn này đến người dùng thứ hai 302b và đợi người dùng thứ hai 302b xác nhận cảnh báo này (các tin nhắn 426). Ngay khi người dùng thứ hai 302b xác nhận cảnh báo, máy chủ thứ hai 104b theo đó thay đổi các trạng thái thành “AlarmStateAcknowledged” (tin nhắn 428), và sau đó gửi tin nhắn đến Vòng Đồng bộ ServerState sao cho hai máy chủ khác 104a,c cũng thay đổi trạng thái tương ứng (khung 430). Máy chủ thứ hai 104b sau đó thay đổi trạng thái thành “NotifyAlarmAcknowledged” (tin nhắn 432) và gửi tin nhắn đến máy chủ thứ ba 104c qua Vòng Đồng bộ ClientNotification để làm cho nó thay đổi trạng thái tương ứng (khung 434). Sau đó, máy chủ thứ ba 104c thông báo cho máy khách 102c rằng hệ thống 100 vừa xác nhận cảnh báo (tin nhắn 436), mà sắp xếp tin nhắn đến người dùng thứ hai 302b (tin nhắn 438).

Theo phương án thay thế (không được mô tả), trong đó máy chủ thứ hai 104b hỏng và có thể không đóng vai trò là chủ cho Vòng Đồng bộ, hệ thống 100 tự chọn một trong số các máy chủ khác 104 đóng vai trò làm chủ cho vòng này. Chủ của vòng đồng bộ chỉ là máy chủ 104 mà cho phép làm cho tất cả các nút khác trên vòng thay đổi trạng thái khi vòng đồng bộ được sử dụng chia sẻ các thông báo trong số các nút.

Fig.7 thể hiện phép hiển thị minh họa 700 được biểu diễn đến người dùng 302 khi xác nhận trạng thái cảnh báo theo sơ đồ 400 của Fig.4. Phép hiển thị 700 bao gồm các bảng video 702a-c (gọi chung là “các bảng 702”) thể hiện video tạo đồng thời gian thực từ máy quay không nút 114; các cảnh báo 704 chỉ ra rằng cảnh báo được xảy ra do máy quay không nút 114 ghi lại; và nút xác nhận 706 mà người dùng thứ hai 302b bấm vào để xác nhận cảnh báo đã xảy ra.

Các hiển thị chia sẻ và ứng dụng hợp tác 222

Các người dùng 302 của hệ thống 100 cũng có thể muốn chia sẻ mỗi hiển thị khác 700 và hợp tác, như bằng cách gửi cho nhau các tin nhắn và nói với nhau qua hệ thống 100, trong khi chia sẻ các hiển thị 700. Các hiển thị được chia sẻ này và ứng dụng hợp tác 222 theo đó cho phép người dùng 302 chia sẻ dữ liệu như trạng thái hiển thị và các thông báo từ máy chủ sang máy khách như các tin nhắn người dùng và chia sẻ các yêu cầu. Loại dữ liệu này là dữ liệu đồng bộ mà được chia sẻ theo thời gian thực.

Fig.5 thể hiện sơ đồ thứ tự UML 500, trong đó các phép hiển thị 700 được chia sẻ giữa các người dùng 302 sử dụng giao thức xử lý đồng thời 214. Sơ đồ 500 bao gồm sáu đối tượng: người dùng thứ nhất và thứ hai 302a,b, máy khách thứ nhất và thứ hai 102a,b mà người dùng thứ nhất và thứ hai 302a,b lần lượt được kết nối, và máy chủ thứ nhất và thứ hai 104a,b mà máy khách thứ nhất và thứ hai 102a,b lần lượt được kết nối.

Người dùng thứ nhất 302a đăng nhập vào máy chủ thứ nhất 104a qua máy khách thứ nhất 102a (tin nhắn 502), sau đó máy chủ thứ nhất 104a tham gia vào Vòng Đồng bộ ClientNotification (khung 504). Tương tự, người dùng thứ hai 302b đăng nhập vào máy chủ thứ hai 104b qua máy khách thứ hai 102b (tin nhắn 506), sau đó máy chủ thứ hai 104b cũng tham gia vào Vòng Đồng bộ ClientNotification (khung 508).

Sau đó, người dùng thứ nhất 302a ra lệnh cho máy khách thứ nhất 102a mà anh ta muốn chia sẻ hiển thị 700. Người dùng thứ nhất 302a thực hiện việc này bằng cách bấm vào nút chia sẻ (tin nhắn 510), làm cho máy khách thứ nhất 102a mở hiển thị 700 được chia sẻ (“hiển thị 700 được chia sẻ”) trên máy chủ thứ nhất 104a (tin nhắn 512). Máy chủ thứ nhất 104a tạo ra phiên hiển thị được chia sẻ (tin nhắn 514), và tiếp đó gửi mã nhận dạng phiên đến máy khách thứ nhất 102a (tin nhắn 516).

Tại một khung 518, một trong số các máy khách 102 tham gia vào Vòng Đồng bộ cho phép chúng chia sẻ hiển thị 700 được chia sẻ. Máy chủ thứ nhất 104a tham gia vào Vòng Đồng bộ SharedView1 ở khung 520. Đồng thời, máy khách thứ nhất 106a ra lệnh cho máy chủ thứ nhất 104a thông báo đến máy chủ khác 104b qua giao thức xử lý đồng thời 214 mà hiển thị của người dùng thứ nhất 302a 700 có thể được chia sẻ bằng cách gửi đến máy chủ thứ nhất 104a danh sách người dùng và mã nhận dạng phiên (tin nhắn 522). Máy chủ thứ nhất 104a làm việc này bằng cách gửi tin nhắn đến máy chủ thứ hai 104b qua Vòng Đồng bộ clientNotify mà làm cho máy chủ thứ hai 104b thay đổi thành trạng thái NotifyViewSession (khung 524). Trong trạng thái NotifyViewSession, máy chủ thứ hai 104b làm cho máy khách thứ hai 106b đẩy nhanh người dùng thứ hai 302b chia sẻ hiển thị của người dùng thứ nhất 302a 700 (các tin nhắn 526 đến 528), và phản ứng khẳng định của người dùng thứ hai 302b được chuyển ngược lại máy chủ thứ

hai 104b (các tin nhắn 530 và 532). Máy chủ thứ hai 104b sau đó tham gia vào Vòng Đồng bộ SharedView1 (khung 534), mà được sử dụng để chia sẻ hiển thị 700 của người dùng thứ nhất 302a.

Tại khung thứ hai 519, mỗi người dùng 106 cập nhật hiển thị 700 được chia sẻ, và các cập nhật tự đồng được chia sẻ với nhau. Người dùng thứ nhất 302a phóng to bảng thứ nhất 702a trong hiển thị 700 được chia sẻ (tin nhắn 536), và máy khách thứ nhất 102a tiếp đến máy chủ thứ nhất 104a làm thế nào người dùng thứ nhất 302a được phóng to thành bảng thứ nhất 702a (tin nhắn 538). Máy chủ thứ nhất 104a chia sẻ các thông tin cụ thể về phóng với máy chủ thứ hai 104b bằng cách cho chúng qua Vòng Đồng bộ SharedView1 (khung 540). Máy chủ thứ hai 104b theo đó cập nhật phép hiển thị 700 được chia sẻ như được hiển thị trên máy khách thứ hai 106b (tin nhắn 542), và hiển thị 700 được chia sẻ được cập nhật tiếp đó được hiển thị cho người dùng thứ hai 302b (tin nhắn 544). Đồng thời, người dùng thứ hai 302b cho ra bảng thứ hai 702b về hiển thị 700 được chia sẻ (tin nhắn 546), và máy khách thứ hai 102b tiếp đến máy chủ thứ hai 104b làm thế nào người dùng thứ hai 302b cho ra bảng này 702b (tin nhắn 548). Tiếp đó, máy chủ thứ hai 104b chia sẻ các thông tin cụ thể cho ra với máy chủ thứ nhất 104a bằng cách cho chúng qua bằng cách sử dụng Vòng Đồng bộ SharedView1 (khung 550). Do đó, máy chủ thứ nhất 104a cập nhật hiển thị 700 được chia sẻ như được thể hiện trên máy khách thứ nhất 106b (tin nhắn 552), và hiển thị chia sẻ được cập nhật 700 tiếp đó được hiển thị cho người dùng thứ nhất 302a (tin nhắn 556).

Sau khung thứ hai 519, người dùng thứ nhất 302a đóng hiển thị 700 (tin nhắn 556), tiếp đến máy chủ thứ nhất 104a (tin nhắn 558). Do đó, máy chủ thứ nhất 104a rời Vòng Đồng bộ SharedView1 (tin nhắn và khung 560). Người dùng thứ hai 302b đóng tương tự hiển thị 700, làm cho máy chủ thứ hai 104b rời Vòng Đồng bộ SharedView1 (các tin nhắn 562 và 564, và tin nhắn và khung 566).

Trong ví dụ của Fig.5, các người dùng 302 cho ra và phóng to hiển thị 700 được chia sẻ. Theo các phương án thay thế (không được mô tả), các người dùng 302 có thể biến đổi hiển thị 700 được chia sẻ theo các cách khác. Ví dụ, mỗi người dùng 302 có thể thay đổi thiết kế của các bảng 702; chọn liệu video có được hiển thị trực tiếp hoặc

theo cách phát lại, trong đó trường hợp các người dùng 302 cũng có thể tạm dừng, chơi, hoặc nhảy qua video; và hiển thị đối tượng người dùng như các bản đồ hoặc các trang web cùng với thông tin về đối tượng người dùng như lược sử sửa đổi. Theo các phương án thay thế này, các ví dụ về thông trạng thái bổ sung mà được đồng bộ hóa bằng cách sử dụng Vòng Đồng bộ bao gồm liệu video có được chạy, tạm dừng, hoặc nhảy qua và lược sử sửa đổi của đối tượng người dùng.

Trong khi phân thảo luận trên đây tập trung vào việc thực hiện các hiển thị được chia sẻ và ứng dụng hợp tác 222 trong hệ thống bảo mật vật lý ngang hàng 100 của Fig.1, nhìn chung, ứng dụng này 222 có thể được thực hiện trong hệ thống bảo mật vật lý mà có nhiều máy chủ 104, như hệ thống trung ương mà bao gồm máy chủ cổng kết nối được tập trung. Ví dụ về phương án chung này được thể hiện trên Fig.12, mà mô tả phương pháp minh họa 1200 để chia sẻ hiển thị sử dụng hệ thống bảo mật vật lý that bao gồm nhiều nút máy chủ. Phương pháp 1200 bắt đầu ở khối 1202 và tiến hành phong tỏa 1204, trong đó dữ liệu trạng thái hiển thị biểu diễn hiển thị được hiển thị bởi máy khách thứ nhất (như máy khách thứ nhất 102a), mà là hiển thị được chia sẻ, được gửi từ máy khách thứ nhất đến nút máy chủ thứ nhất (như máy chủ thứ nhất 104a và dữ liệu trạng thái hiển thị được gửi qua tin nhắn 538). Ở khối 1206, dữ liệu trạng thái hiển thị tiếp đến từ nút máy chủ thứ nhất đến máy khách thứ hai (như máy khách thứ hai 102b) qua nút máy chủ thứ hai (như máy chủ thứ hai 104b và dữ liệu trạng thái hiển thị được gửi qua khung 540 và tin nhắn 542). Ở khối 1208, máy khách thứ hai tiếp đó cập nhật màn hình sử dụng dữ liệu trạng thái hiển thị để thể hiện hiển thị được chia sẻ (như qua tin nhắn 544). Trả lời lại sự thay đổi về hiển thị được chia sẻ ở máy khách thứ hai, như sự thay đổi do sự tương tác với người dùng ở máy khách thứ hai (như qua tin nhắn 546), ở khối 1210 được cập nhật dữ liệu trạng thái hiển thị được gửi từ máy khách thứ hai đến nút máy chủ thứ hai (như qua tin nhắn 548). Trạng thái hiển thị được cập nhật biểu diễn hiển thị được chia sẻ như được hiển thị bằng máy khách thứ hai. Trạng thái hiển thị được cập nhật được gửi từ nút máy chủ thứ hai đến máy khách thứ nhất qua nút máy chủ thứ nhất tại khối 1212 (như qua khung 550 và tin nhắn 552), và tại khối 1214, màn hình của máy khách thứ nhất tiếp đó được cập nhật để thể hiện hiển thị được chia sẻ do nó được biến đổi ở máy khách thứ hai sử dụng trạng thái hiển thị được cập nhật (như qua tin nhắn 554). Phương pháp 1200 kết

thúc ở khối 1216. Theo phương án thay thế, như khi thực hiện hệ thống trung ương mà sử dụng máy chủ cổng kết nối được tập trung, tất cả dữ liệu trạng thái hiển thị có thể được định tuyến qua máy chủ được tập trung.

Ứng dụng chia sẻ hiển thị không được giám sát 225

Các người dùng 302 của hệ thống 100 cũng có thể muốn xem và điều khiển hiển thị trên màn hình mà được kết nối trực tiếp hoặc gián tiếp với một trong số các máy chủ 104 mà các người dùng 302 không điều khiển trực tiếp (nghĩa là các người dùng 302 điều khiển qua các máy chủ khác 104) (màn hình này là “màn hình không được giám sát”, và hiển thị trên màn hình không được giám sát là “hiển thị không được giám sát”). Ví dụ, màn hình không được giám sát may được lắp trên tường trước các người dùng 302 và được nối với cụm máy chủ 108 qua một trong số các máy chủ 104 trong cụm 108, trong khi các máy chủ 302 có thể được nối với cụm máy chủ 108 qua các máy chủ khác 104 trong cụm 108. Như được thảo luận sau đây đối với Fig.10, ứng dụng chia sẻ hiển thị không được giám sát 225 cho phép các người dùng 302 hiển thị và điều khiển hiển thị không được giám sát mặc dù không có người dùng nào 302 được kết nối trực tiếp với máy chủ 104 điều khiển hiển thị không được giám sát. Dữ liệu hiển thị được trao đổi giữa các máy chủ 104 cho phép chức năng này là Dữ liệu đồng bộ được chia sẻ theo thời gian thực.

Fig.10 thể hiện Sơ đồ thứ tự UML 1000, trong đó hiển thị không được giám sát được chia sẻ với người dùng thứ nhất 302a sử dụng giao thức xử lý đồng thời 214. Sơ đồ 1000 bao gồm sáu đối tượng: người dùng thứ nhất 302a, máy khách thứ nhất 102a mà người dùng thứ nhất 302a được kết nối và bao gồm màn hình (“màn hình máy khách”) mà người dùng thứ nhất 302a tương tác với, máy chủ thứ nhất và thứ hai 104a,b, đối tượng kiểm soát 1004 chạy trên phần cứng như một trong số các máy khách không được giám sát 102 được kết nối với cả hai máy chủ thứ hai 104b và màn hình không được giám sát, và quản trị 1002 cài đặt đối tượng kiểm soát 1004. Theo phương án thay thế (không được mô tả), màn hình không được giám sát có thể được kết nối trực tiếp với máy chủ thứ hai 104b và đối tượng kiểm soát 1004 có thể chạy trên máy chủ thứ hai 104b.

Trên Fig.10, quản trị 1002 tạo ra đối tượng kiểm soát 1004 (tin nhắn 1006) và đối tượng kiểm soát 1004 tiếp đó tự động đăng nhập vào máy chủ thứ hai 104b (các tin nhắn 1008 và 1010). Đối tượng kiểm soát 1004 làm cho hiển thị không được giám sát có sẵn với máy chủ thứ hai 104b bằng cách gọi SharedViewOpen(viewState) trên máy chủ thứ hai 104, trong đó viewState là dữ liệu trạng thái hiển thị chỉ dẫn hiển thị không được giám sát (tin nhắn 1012). Sau đó, máy chủ thứ hai này 104b tạo ra phiên hiển thị được chia sẻ (tin nhắn 1014) bằng cách chạy SharedViewSessionCreate() và tiếp đó gửi mã nhận dạng phiên tương ứng đến đối tượng kiểm soát (tin nhắn 1016). Sau khi nhận mã nhận dạng phiên đối tượng kiểm soát 1004 tham gia vào Vòng Đồng bộ SharedView1 (khung 1018), mà được sử dụng để truyền which dữ liệu trạng thái hiển thị đến và từ các máy chủ khác 104 trong cụm 108 mà cũng là các thành phần của Vòng Đồng bộ SharedView1.

Sau khi tham gia vào Vòng Đồng bộ SharedView1, đối tượng kiểm soát 1020 công bố thông báo đến các máy chủ khác 104 trong cụm 108 mà hiển thị không được giám sát có sẵn được nhìn thấy và được điều khiển. Đối tượng kiểm soát 1020 làm việc này bằng cách gọi RegisterMonitor(sessionId) trên máy chủ thứ hai 104b (tin nhắn 1018), làm cho mã nhận dạng phiên liên quan đến hiển thị không được giám sát được đăng ký trong thư mục hiển thị (khung 1022). Thư mục hiển thị được chia sẻ với các máy chủ khác 104 trong cụm 108 sử dụng Giao thức nhất quán 216.

Ngay khi thư mục hiển thị được truyền đến các máy chủ khác 104 trong cụm 108, các máy chủ khác đó 104 có thể truy cập thư mục hiển thị để xác định các hiển thị không được giám sát nào có sẵn để hiển thị và điều khiển. Sau khi máy chủ thứ nhất 104a nhận thư mục hiển thị, người dùng thứ nhất 302a qua máy khách thứ nhất 102a đăng nhập vào máy chủ thứ nhất 104a, theo đó truy cập vào cụm 108 (các tin nhắn 1024) và thư mục hiển thị. Người dùng thứ nhất 102a ra lệnh cho máy khách thứ nhất 102a để hiển thị hiển thị không được giám sát bằng cách gọi UIDisplayMonitor(sessionId) (tin nhắn 1026), làm cho máy khách thứ nhất 102a gửi mã nhận dạng phiên hiển thị không được giám sát đến máy chủ thứ nhất 104a với các lệnh để mở hiển thị không được giám sát (tin nhắn 1028). Máy chủ thứ nhất 104a xác nhận các lệnh của máy khách thứ nhất 102a (tin nhắn 1030) và tiếp đó tham gia vào Vòng Đồng bộ SharedView1 (khung 1032) để tự động nhận dữ liệu trạng thái hiển

thị mô tả hiển thị hiện thời của màn hình không được giám sát (tin nhắn 1034) và tự động giữ thông tin các thay đổi sau đó bất kỳ đến hiển thị không được giám sát.

Người dùng thứ nhất 302a sau đó cho ra một trong số các bảng của hiển thị không được giám sát do nó được hiển thị trên màn hình máy khách (tin nhắn 1036), và máy khách thứ nhất 102a tiếp đến hành động cho ra và sự tương đồng của bảng cụ thể mà được cho ra vào máy chủ thứ nhất 104a bằng cách gọi `SharedViewUpdate(action=pan, panelId=2)` (tin nhắn 1038). Máy chủ thứ nhất 104a gửi dữ liệu trạng thái hiển thị được cập nhật đến tất cả các máy chủ 104 là các thành phần của Vòng Đồng bộ `SharedView1` (khung 1040), mà cho phép tất cả các máy chủ 104 để sao chép phiên bản của hiển thị không được giám sát. Máy chủ thứ hai 104b nhận dữ liệu trạng thái hiển thị được cập nhật và tiếp nó đến đối tượng kiểm soát 1004 bằng cách gọi `NotifySharedViewUpdate(action=pan, params, panelId=2)` (tin nhắn 1042). Đối tượng kiểm soát 1004 tiếp đó cập nhật màn hình không được giám sát để thể hiện hiển thị không được giám sát như được biến đổi bởi người dùng thứ nhất 302a (tin nhắn 1044).

Trong ví dụ của Fig.10, người dùng thứ nhất 302a cho ra một trong số các bảng của hiển thị không được giám sát. Theo các phương án thay thế (không được mô tả), người dùng thứ nhất 302a có thể biến đổi hiển thị không được giám sát theo các cách khác. Ví dụ, người dùng thứ nhất 302a có thể thay đổi thiết kế của một hoặc nhiều bảng của hiển thị không được giám sát; chọn liệu video có được hiển thị trực tiếp hoặc theo cách phát lại, trong trường hợp mà người dùng thứ nhất 302a cũng có thể dừng, chạy, hoặc nhảy qua video; và hiển thị các đối tượng người dùng bản đồ hoặc các trang web cùng với thông tin về đối tượng người dùng như lược sử sửa đổi. Theo các phương án thay thế này, các ví dụ về thông tin trạng thái bổ sung mà được đồng bộ hóa sử dụng Vòng Đồng bộ bao gồm liệu video đang chạy, dừng, hoặc nhảy qua và lược sử sửa đổi của đối tượng người dùng.

Theo phương án thay thế khác (không được mô tả), ứng dụng chia sẻ hiển thị không được giám sát 225 có thể được sử dụng để tạo ra màn hình hội tụ bao gồm ma trận của $n \times m$ màn hình không được giám sát. Ví dụ, trong đó $n = m = 2$ và do đó, có bốn màn hình không được giám sát, người dùng thứ nhất 302a có thể điều khiển tất cả

bốn màn hình không được giám sát đồng thời để tạo ra một màn hình ảo, lớn. Sau đó, một video có thể được phóng to sao cho một trong số các hiển thị không được giám sát là một phần tư của video, theo đó cho phép video được phóng to và thể hiện qua bốn màn hình không được giám sát. Theo phương án này, các đối tượng kiểm soát 1004 cho màn hình không được giám sát có thể giao tiếp với cụm máy chủ 108 qua một đến bốn máy chủ bất kỳ 104.

Trong khi Fig.10 chỉ thể hiện người dùng thứ nhất 302a, theo các phương án thay thế (không được mô tả), nhiều hơn một người dùng 302 có thể xem và điều khiển hiển thị không được giám sát cũng bằng cách tham gia vào Vòng Đồng bộ SharedView1. Trong các ví dụ trên đây của màn hình hội tụ bao gồm màn trận $n \times m$ màn hình không được giám sát, màn hình hội tụ có thể được lắp trong phòng để xem đồng thời vài người dùng 302 với một trong số các người dùng 302 có khả năng điều khiển một trong số các hiển thị không được giám sát.

Trong khi phần thảo luận trên đây tập trung vào sự thực hiện của ứng dụng chia sẻ hiển thị không được giám sát 225 trong hệ thống bảo mật vật lý ngang hàng 100 của Fig.1, nói chung, ứng dụng này 225 có thể được thực hiện trong hệ thống bảo mật vật lý có nhiều máy chủ 104, như hệ thống trung ương bao gồm máy chủ cổng kết nối được tập trung. Ví dụ về phương án chung hơn này được thể hiện trên Fig.11, mà mô tả phương pháp minh họa 1100 để tương tác với màn hình không được giám sát trong hệ thống bảo mật vật lý bao gồm nhiều nút máy chủ. Phương pháp này bắt đầu ở khối 1102 và tiến hành phong tỏa 1104, trong đó nút máy chủ thứ hai (như máy chủ thứ hai 104b) giao tiếp với màn hình không được giám sát gửi đến nút máy chủ thứ nhất (như máy chủ thứ nhất 104a) dữ liệu trạng thái hiển thị chỉ dẫn hiển thị không được giám sát (như qua Vòng Đồng bộ ở các khung 1020 và 1032 của Fig.10). Phương pháp 1100 tiếp đó tiến hành phong tỏa 1106, trong đó ít nhất một phần của hiển thị không được giám sát được hiển thị trên màn hình máy khách (như cập nhật màn hình máy khách do tin nhắn 1034 của Fig.10) mà theo đó phương pháp 1100 kết thúc ở khối 1108. Theo phương án thay thế, như khi tiến hành với hệ thống trung ương sử dụng máy chủ cổng kết nối được tập trung, tất cả dữ liệu trạng thái hiển thị có thể được định tuyến qua máy chủ được tập trung đó.

Ứng dụng các dòng cụm 220

Một trong số các người dùng 302 cũng có thể muốn video dòng từ một trong số các máy quay 106,114 nếu kết nối điểm với điểm giữa người dùng 302 và máy quay 106,114 không có sẵn; ứng dụng các dòng cụm 220 cho phép chức năng này. Fig.6 thể hiện Sơ đồ thứ tự UML 600, trong đó video được tạo dòng từ máy quay không nút 114 đến người dùng thứ nhất 302a qua máy chủ thứ nhất và thứ hai 104a,b và máy khách thứ nhất 102a. Sơ đồ UML có năm đối tượng: người dùng thứ nhất 302a, máy khách thứ nhất 102a, máy chủ thứ nhất và thứ hai 104a,b, và máy quay không nút 114. Máy khách thứ nhất 102a có thể giao tiếp trực tiếp với máy chủ thứ nhất 104a, nhưng không thể giao tiếp trực tiếp với máy chủ thứ hai 104b. Tuy nhiên, máy chủ thứ nhất và thứ hai 104a,b có thể giao tiếp trực tiếp với nhau. Ngoài ra, trong khi máy chủ thứ hai 104b và máy quay không nút 114 có thể giao tiếp với nhau, máy chủ thứ nhất 104a và máy quay không nút 114 không thể giao tiếp trực tiếp.

Máy chủ thứ hai 104b trước hết tạo ra phiên với máy quay không nút 114 sao cho video được tạo dòng từ máy quay không nút 114 đến máy chủ thứ hai 104b. Máy chủ thứ hai 104b trước hết cài đặt phiên Giao thức tạo dòng thời gian thực (RTSP) với máy quay không nút 114 (các tin nhắn 602 và 604), và ra lệnh cho máy quay nút 114 gửi cho nó video (các tin nhắn 606 và 608). Máy quay không nút 114 sau đó bắt đầu tạo dòng (tin nhắn 610).

Người dùng thứ nhất 302a thiết lập kết nối với máy khách thứ nhất 102a (tin nhắn 612) và tiếp đó ra lệnh cho máy khách thứ nhất 102a mở cửa sổ thể hiện video tạo dòng (tin nhắn 614). Máy khách thứ nhất 102a sau đó gọi LookupRoute() xác định máy chủ 104 kết nối với ai (tin nhắn 616); do máy khách thứ nhất 102a không thể kết nối trực tiếp với máy chủ thứ hai 104b, nó cài đặt kết nối RTSP với máy chủ thứ nhất 104a (tin nhắn 618). Sau đó, máy chủ thứ nhất 104b gọi LookupRoute() để xác định nút kết nối để truy cập video thời gian thực, và xác định nó nên kết nối với máy chủ thứ hai 104b (tin nhắn 620). Sau đó, máy chủ thứ nhất 104a cài đặt kết nối RTSP với máy chủ thứ hai 104b (tin nhắn 622), và máy chủ thứ hai 104b trả lại mã nhận dạng phiên về máy chủ thứ nhất 104a (tin nhắn 624). Máy chủ thứ nhất 104a tiếp tục mã nhận dạng phiên đến máy khách thứ nhất 102a (tin nhắn 626). Sử dụng mã nhận dạng phiên này, máy khách thứ nhất 102a ra lệnh cho máy chủ thứ hai 104b bắt đầu

chạy video RTSP (các tin nhắn 628 đến 634), và sau đó, máy chủ thứ hai 104b tạo dòng video đến người dùng thứ nhất 302a qua máy chủ thứ hai 104b, sau đó là máy chủ thứ nhất 104a, và tiếp đó là máy khách thứ nhất 102a (các tin nhắn 636 đến 640).

Trong khi Fig.6 định tuyến video từ một trong số các máy quay không nút 114 được kết nối với một trong số các máy chủ 104 trong cụm 108 đến các máy chủ khác 104 theo cùng cụm 108, theo các phương án thay thế (không được mô tả), video cũng có thể được định tuyến từ một trong số các máy quay nút 106 trong cụm 108 thông qua các máy quay nút khác 106 trong cùng cụm 108.

Khởi động

Theo phương án, thông tin hội viên của cụm được lưu trữ cục bộ lâu dài một cách cục bộ trên một trong số các nút. Khi một trong số các nút này khởi động lại, nó tham gia lại một cách tự động cụm 108 mà nó là thành phần trước khi khởi động lại. Việc này được mô tả trong phương pháp minh họa 900 được thể hiện trên Fig.9. Sau khi tiến hành phong tỏa 806, một trong số các nút trong cụm 108 khởi động lại (khối 902). Dựa trên việc khởi động lại, nút này truy cập vào thông tin hội viên của cụm được lưu trữ theo cách lâu dài mà nhận dạng cụm 108 mà nó là thành phần trước khi khởi động lại (khối 904), và sau đó, tham gia lại cụm này 108 (khối 906) trước khi quay trở lại khối 808. Có các nút tham gia lại một cách tự động vào cụm 108, việc khởi động sau có lợi ở chỗ nó giúp hệ thống 100 phục hồi sau khởi động của một hoặc nhiều máy chủ bất kỳ của nó. Do một trong số các nút lưu trữ vĩnh cửu thông tin nhất quán, dựa trên việc chỉ tham gia lại vào cụm 108 mà thông tin nhất quán thay đổi do nút cuối cùng rời khỏi cụm 108 được đồng bộ hóa lại, theo đó tiết kiệm băng thông.

Trong khi một số phương án minh họa được mô tả, các phương án thay thế, không được mô tả, có thể có. Ví dụ, trong khi phương án được mô tả, các máy quay nút 106 và các máy quay không nút 114 khác biệt với nhau, theo các phương án thay thế (không được mô tả), một máy quay có thể đồng thời là máy quay nút và máy quay không nút. Ví dụ, trên Fig.1, máy quay thứ nhất 106a là nút mà là thành phần của cụm thứ ba 108c; tuy nhiên, nếu máy quay thứ nhất 106a cũng được nối trực tiếp với máy chủ thứ năm 104e nhưng chỉ giữ lại thông tin hội viên của cụm của nó cho cụm thứ ba 108c, máy quay thứ ba 106a sẽ duy trì là thành phần của cụm thứ ba 108c trong khi

đóng vai trò đồng thời là máy quay không nút 114 từ triển vọng của máy chủ thứ năm 104e.

Bộ xử lý được sử dụng trong các phương án trước đây có thể là, ví dụ, bộ vi xử lý, bộ vi điều khiển, bộ điều khiển logic có thể lập trình được, mảng cổng có thể lập trình trường, hoặc mạch tích hợp cụ thể với ứng dụng. Các ví dụ về các vật ghi có thể đọc được bằng máy tính là không tạm thời và bao gồm vật ghi dự trên đĩa các CD-ROM và DVD, vật ghi từ như cá đĩa cứng và các dạng khác lưu trữ đĩa từ, vật ghi dựa trên chất bán dẫn như vật ghi nhanh, bộ nhớ truy cập ngẫu nhiên, và bộ nhớ chỉ đọc.

Dự tính rằng một phần bất kỳ của khía cạnh bất kỳ hoặc phương án được thảo luận trong bản mô tả này có thể được thực hiện hoặc kết hợp với phần bất kỳ của khía cạnh hoặc phương án bất kỳ được thảo luận trong bản mô tả này.

Với mục đích tiện lợi, các phương án minh họa được mô tả làm các khối chức năng được kết nối qua lại khác nhau. Tuy nhiên, điều này không cần thiết và có thể có các trường hợp mà các khối chức năng này tương đương kết hợp với một thiết bị logic, chương trình hoặc hoạt động bằng các biên không rõ ràng. Trong trường hợp bất kỳ, các khối chức năng tự có thể được thực hiện bởi bản thân chúng, hoặc kết hợp với các phần khác của phần cứng hoặc phần mềm.

Trong khi các phương án cụ thể được mô tả trước đây, nên được hiểu rằng các phương án khác cũng có thể có và có bao gồm ở đây. Sẽ rõ ràng đối với chuyên gia trong lĩnh vực này rằng các biến đổi và các điều chỉnh đối với các phương án trước đây, không được thể hiện, có thể thực hiện được.

YÊU CẦU BẢO HỘ

1. Phương pháp chia sẻ dữ liệu trong hệ thống bảo mật vật lý bao gồm nhiều nút máy chủ, phương pháp này bao gồm các bước:

(a) bổ sung nút máy chủ thứ nhất vào cụm máy chủ chứa nút máy chủ thứ hai bằng cách thực hiện phương pháp bao gồm bước:

(i) trao đổi phiên bản của thông tin hội viên của cụm được lưu trữ trên nút máy chủ thứ nhất với phiên bản của thông tin hội viên của cụm được lưu trữ trên một trong số các nút máy chủ hiện đã là một phần của cụm máy chủ; và

(ii) đồng bộ hóa các phiên bản của thông tin hội viên của cụm được lưu trữ trên nút máy chủ thứ nhất với các phiên bản của thông tin hội viên của cụm được lưu trữ trên tất cả các nút máy chủ tạo thành một phần của cụm trước khi nút máy chủ thứ nhất tham gia vào cụm;

(b) truy cập, sau khi nút máy chủ thứ nhất đã được thêm vào cụm máy chủ và sử dụng nút thứ nhất, mã nhận dạng nút nhận dạng nút khác của cụm máy chủ, trong đó mã nhận dạng nút bao gồm ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả các nút máy chủ trong cụm máy chủ và có thể truy cập được bởi tất cả các nút máy chủ trong cụm máy chủ; và trong đó mỗi nút trong các nút máy chủ trong cụm máy chủ lưu trữ theo cách lâu dài phiên bản riêng của nó về thông tin hội viên của cụm một cách cục bộ; và

(c) gửi dữ liệu từ nút thứ nhất đến nút khác.

2. Phương pháp theo điểm 1, trong đó cụm máy chủ bao gồm ít nhất ba nút máy chủ.

3. Phương pháp theo điểm 1 hoặc 2, trong đó các nút máy chủ bao gồm máy quay, thiết bị ghi video mạng, và máy chủ điều khiển truy cập.

4. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 3 còn bao gồm bước:

(a) truy cập, sử dụng nút thứ hai, mã nhận dạng nút nhận dạng nút thứ nhất; và

- (b) gửi dữ liệu bổ sung từ nút thứ hai đến nút thứ nhất.
5. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 4, trong đó thông tin hội viên của cụm bao gồm:
- (a) mã nhận dạng nút nhận dạng duy nhất từng nút trong số các nút máy chủ trong cụm máy chủ; và
 - (b) mã nhận dạng cụm nhận dạng duy nhất cụm máy chủ mà các nút máy chủ thuộc về nó.
6. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 5 còn bao gồm bước:
- (a) khởi động lại một trong số các nút máy chủ (“nút máy chủ được khởi động lại”) trong cụm máy chủ; và
 - (b) ngay khi nút máy chủ được khởi động lại quay trở lại trực tuyến, sử dụng nút máy chủ được khởi động lại để thực hiện phương pháp bao gồm bước:
 - (i) truy cập mã nhận dạng cụm nhận dạng cụm máy chủ; và
 - (ii) tham gia lại một cách tự động vào cụm máy chủ.
7. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 6, trong đó bước gửi dữ liệu bao gồm:
- (a) tìm kiếm, sử dụng nút thứ nhất, điểm cuối giao tiếp cho nút thứ hai từ mã nhận dạng nút; và
 - (b) gửi dữ liệu từ nút thứ nhất đến điểm cuối giao tiếp.
8. Phương pháp theo điểm 7, trong đó điểm cuối giao tiếp và mã nhận dạng nút bao gồm các mục nhập trong mã nhận dạng nút liên quan đến bản đồ mạng cho tất cả các nút máy chủ trong cụm máy chủ đến các điểm cuối giao tiếp tương ứng, và trong đó từng nút trong số các nút máy chủ trong cụm máy chủ lưu trữ theo cách lâu dài phiên bản riêng của nó về bản đồ mạng một cách cục bộ.
9. Phương pháp theo điểm 8, trong đó bản đồ mạng cho phép từng nút trong số các nút máy chủ trong cụm máy chủ gửi dữ liệu đến nút bất kỳ khác trong số các nút máy chủ trong cụm máy chủ mà không sử dụng máy chủ được tập trung.

10. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 9, trong đó dữ liệu được lưu trữ cục bộ trên nút thứ nhất và còn bao gồm việc biến đổi dữ liệu sử dụng nút thứ nhất, trong đó việc gửi dữ liệu từ nút thứ nhất đến nút thứ hai bao gồm một phần đồng bộ hóa dữ liệu trên nút thứ nhất và thứ hai sau khi nút thứ nhất biến đổi dữ liệu.

11. Phương pháp theo điểm 10, trong đó dữ liệu bao gồm thông tin phiên bản được tạo ra bằng cách sử dụng cơ chế tạo phiên bản mang tính nhân quả và các phiên bản khác nhau của dữ liệu được lưu trữ trên nút thứ nhất và thứ hai, và trong đó việc đồng bộ hóa dữ liệu bao gồm việc so sánh thông tin phiên bản được lưu trữ trên nút thứ nhất và thứ hai và chấp nhận trên cả hai nút thứ nhất và thứ hai dữ liệu có thông tin phiên bản chỉ ra là gần hơn.

12. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 11, trong đó dữ liệu bao gồm mã nhận dạng nút của nút thứ nhất, thông tin trạng thái nhịp của nút thứ nhất, thông tin trạng thái ứng dụng của nút thứ nhất, và thông tin phiên bản, và trong đó việc gửi dữ liệu bao gồm truyền dữ liệu đến tất cả các nút máy chủ trong cụm máy chủ sử dụng giao thức tán gẫu thực hiện các trao đổi dữ liệu giữa các cặp nút máy chủ trong cụm.

13. Phương pháp theo điểm 12, trong đó dữ liệu được truyền một cách định kỳ đến tất cả nút máy chủ trong cụm máy chủ.

14. Phương pháp theo điểm 12 hoặc 13, trong đó dữ liệu được gửi đến nút thứ hai khi nút thứ nhất tham gia vào cụm.

15. Phương pháp theo điểm bất kỳ trong số các điểm từ 12 đến 14, trong đó miền tập trung các mục nhập mà có thể được biến đổi bằng nút bất kỳ của các nút máy chủ bất kỳ trong cụm máy chủ được lưu trữ cục bộ trên từng nút trong số các nút trong cụm máy chủ, và còn bao gồm việc tạo thông tin phiên bản sử dụng cơ chế tạo phiên bản mang tính nhân quả sao cho thông tin phiên bản chỉ ra rằng nút nào trong các nút máy chủ là nút được biến đổi một cách gần nhất trong số các mục nhập.

16. Phương pháp theo điểm 15, trong đó thông tin trạng thái ứng dụng bao gồm thông tin băm mức đỉnh được tạo ra bằng cách băm tất cả các dữ liệu mục nhập trong miền.

17. Phương pháp theo điểm 16 còn bao gồm bước:

(a) so sánh, sử dụng nút thứ hai, thông tin băm mức đỉnh với thông tin băm mức đỉnh được tạo ra bằng cách băm phiên bản của miền tương ứng được lưu trữ cục bộ trên nút thứ hai; và

(b) nếu các thông tin băm mức đỉnh khác nhau, thì việc đồng bộ hóa các miền trên cả hai nút thứ nhất và thứ hai sẽ sử dụng thông tin phiên bản.

18. Phương pháp theo điểm bất kỳ trong số các điểm từ 12 đến 17, trong đó mục nhập trạng thái chỉ có thể được biến đổi bằng nút thứ nhất được lưu trữ cục bộ trên nút thứ nhất, và trong đó thông tin phiên bản bao gồm số phiên bản mà nút thứ nhất gia tăng bất kỳ khi nào nó biến đổi mục nhập trạng thái.

19. Phương pháp theo điểm 18, trong đó thông tin trạng thái ứng dụng bao gồm cặp thực thể trạng thái bao gồm mã nhận dạng thực thể trạng thái nhận dạng mục nhập trạng thái và số phiên bản.

20. Phương pháp theo điểm 19 còn bao gồm bước:

(a) so sánh, sử dụng nút thứ hai, số phiên bản nhận được từ nút thứ nhất với số phiên bản của mục nhập trạng thái tương ứng được lưu trữ cục bộ trên nút thứ hai; và

(b) nếu các số phiên bản khác nhau, thì cập nhật mục nhập trạng thái được lưu trữ cục bộ trên nút thứ hai với mục nhập trạng thái được lưu trữ cục bộ trên nút thứ nhất.

21. Phương pháp theo điểm 20, trong đó bước cập nhật mục nhập trạng thái bao gồm việc gửi từ nút thứ nhất đến nút thứ hai các mục nhập trạng thái bổ sung được lưu trữ cục bộ trên nút thứ nhất mà được biến đổi đồng thời với mục nhập trạng thái.

22. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 21, trong đó nút thứ nhất và thứ hai bao gồm ít nhất một phần của nhóm nút máy chủ trong cụm mà nút thứ nhất có thể gửi dữ liệu theo cách theo thứ tự tổng thể đến tất cả các nút máy chủ trong nhóm, và trong đó việc gửi dữ liệu bao gồm nút thứ nhất gửi dữ liệu đến tất cả các nút máy chủ trong nhóm.

23. Phương pháp theo điểm 22, trong đó dữ liệu bao gồm dữ liệu không lâu dài được tạo ra trong thời gian chạy của hệ thống bảo mật vật lý.

24. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 23, trong đó dữ liệu bao gồm video tạo dòng được tạo dòng từ nút khác trong số các nút máy chủ trong cụm máy chủ từ nút thứ nhất đến nút thứ hai.

25. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 24, trong đó nút khác là nút thứ hai.

26. Hệ thống bảo mật vật lý bao gồm nhiều nút máy chủ, trong đó hệ thống bảo mật vật lý được định cấu hình để chia sẻ dữ liệu bằng cách thực hiện phương pháp bao gồm các bước:

(a) bổ sung nút máy chủ thứ nhất vào cụm máy chủ chứa nút máy chủ thứ hai, việc bổ sung bao gồm:

(i) trao đổi phiên bản của thông tin hội viên của cụm được lưu trữ trên nút máy chủ thứ nhất với phiên bản của thông tin hội viên của cụm được lưu trữ trên một trong số các nút máy chủ hiện đã là một phần của cụm máy chủ; và

(ii) đồng bộ hóa phiên bản của thông tin hội viên của cụm được lưu trữ trên nút máy chủ thứ nhất với các phiên bản của thông tin hội viên của cụm được lưu trữ trên tất cả các nút máy chủ tạo thành một phần của cụm trước khi nút máy chủ thứ nhất tham gia vào cụm;

(b) truy cập, sau khi nút máy chủ thứ nhất đã được thêm vào cụm máy chủ và sử dụng nút thứ nhất, mã nhận dạng nút nhận dạng nút khác của cụm máy chủ, trong đó mã nhận dạng nút bao gồm ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả các nút máy chủ trong cụm máy chủ và có thể truy cập được bởi tất cả các nút máy chủ trong cụm máy chủ, và trong đó mỗi nút trong các nút máy chủ trong cụm máy chủ lưu trữ theo cách lâu dài phiên bản riêng của nó về thông tin hội viên của cụm một cách cục bộ; và

(c) gửi dữ liệu từ nút thứ nhất đến nút khác.

27. Vật ghi bất biến đọc được bằng máy tính mà các câu lệnh và các chỉ dẫn được ghi mã trên đó làm cho một hoặc nhiều bộ xử lý thực hiện phương pháp chia sẻ dữ liệu trong hệ thống bảo mật vật lý mà bao gồm nhiều nút máy chủ, phương pháp này bao gồm các bước:

(a) bổ sung nút máy chủ thứ nhất vào cụm máy chủ chứa nút máy chủ thứ hai, việc bổ sung bao gồm:

(i) trao đổi phiên bản của thông tin hội viên của cụm được lưu trữ trên nút máy chủ thứ nhất với phiên bản của thông tin hội viên của cụm được lưu trữ trên một trong số các nút máy chủ hiện đã là một phần của cụm máy chủ; và

(ii) đồng bộ hóa các phiên bản của thông tin hội viên của cụm được lưu trữ trên nút máy chủ thứ nhất với các phiên bản của thông tin hội viên của cụm được lưu trữ trên tất cả các nút máy chủ tạo thành một phần của cụm trước khi nút máy chủ thứ nhất tham gia vào cụm;

(b) truy cập, sau khi nút máy chủ thứ nhất đã được thêm vào cụm máy chủ và sử dụng nút thứ nhất, mã nhận dạng nút nhận dạng nút khác của cụm máy chủ, trong đó mã nhận dạng nút bao gồm ít nhất một phần của thông tin hội viên của cụm nhận dạng tất cả các nút máy chủ trong cụm máy chủ và có thể truy cập được bởi tất cả các nút máy chủ trong cụm máy chủ, và trong đó mỗi nút trong các nút máy chủ trong cụm máy chủ lưu trữ theo cách lâu dài phiên bản riêng của nó về thông tin hội viên của cụm một cách cục bộ; và

(c) gửi dữ liệu từ nút thứ nhất đến nút khác.

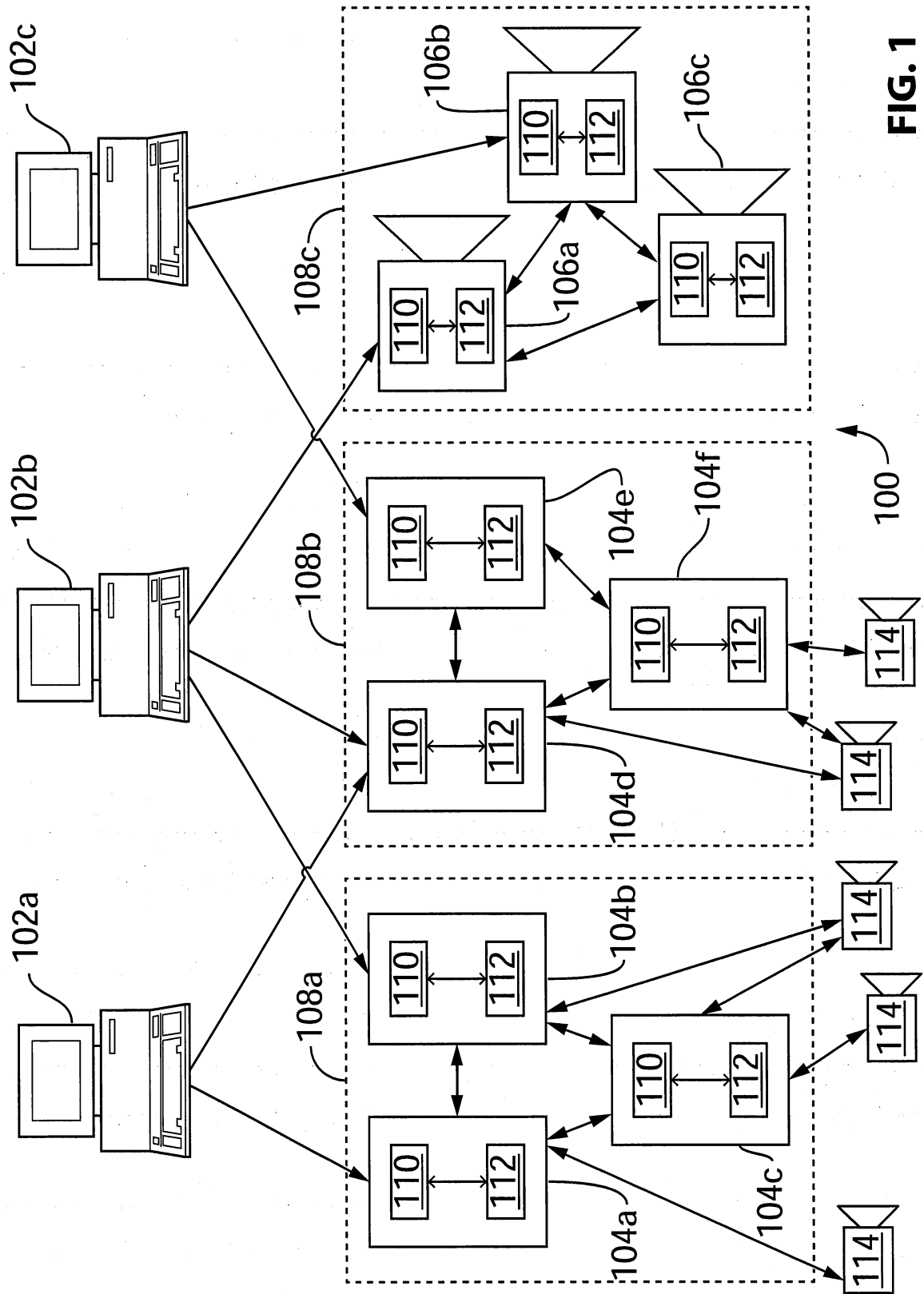


FIG. 1

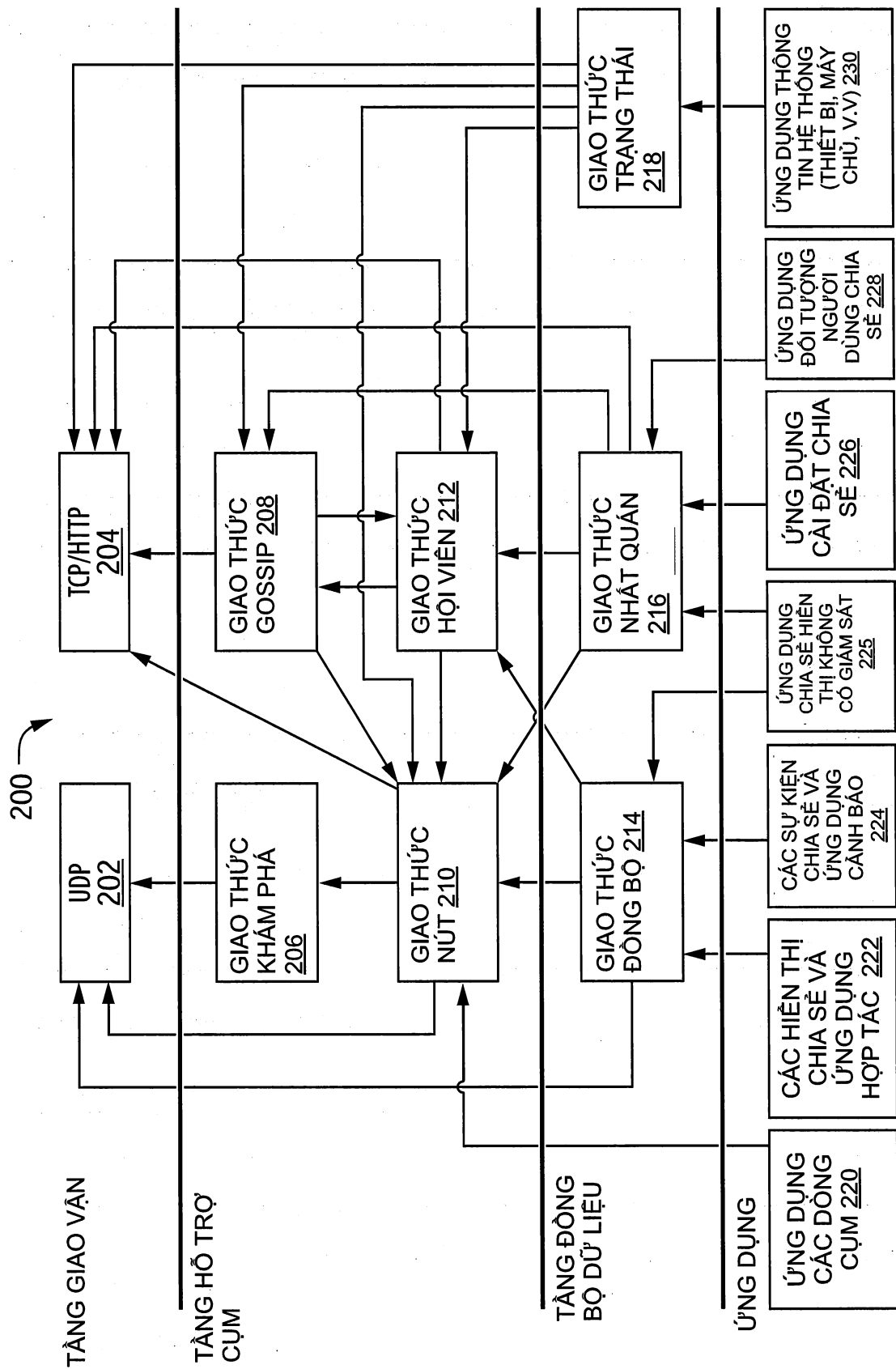
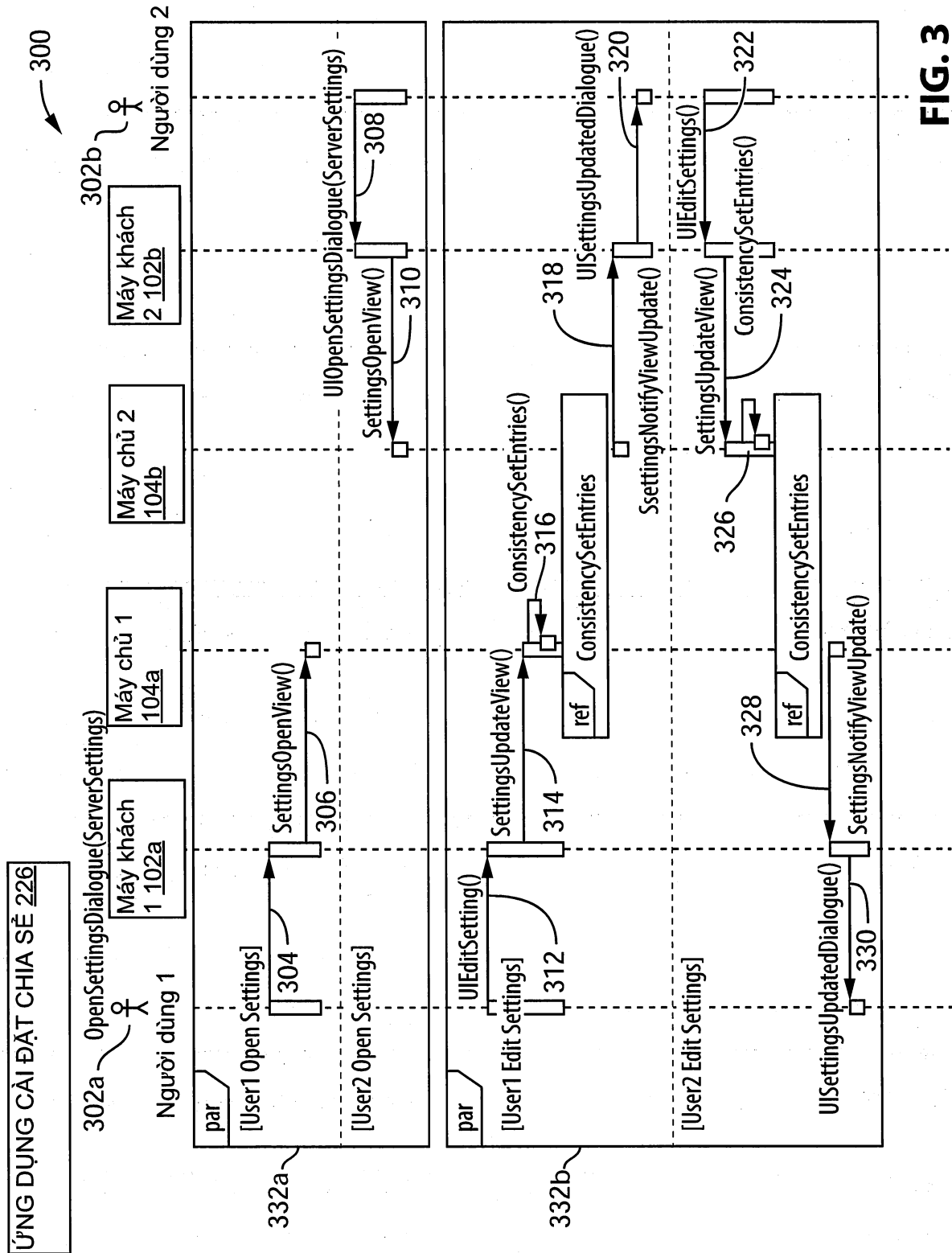
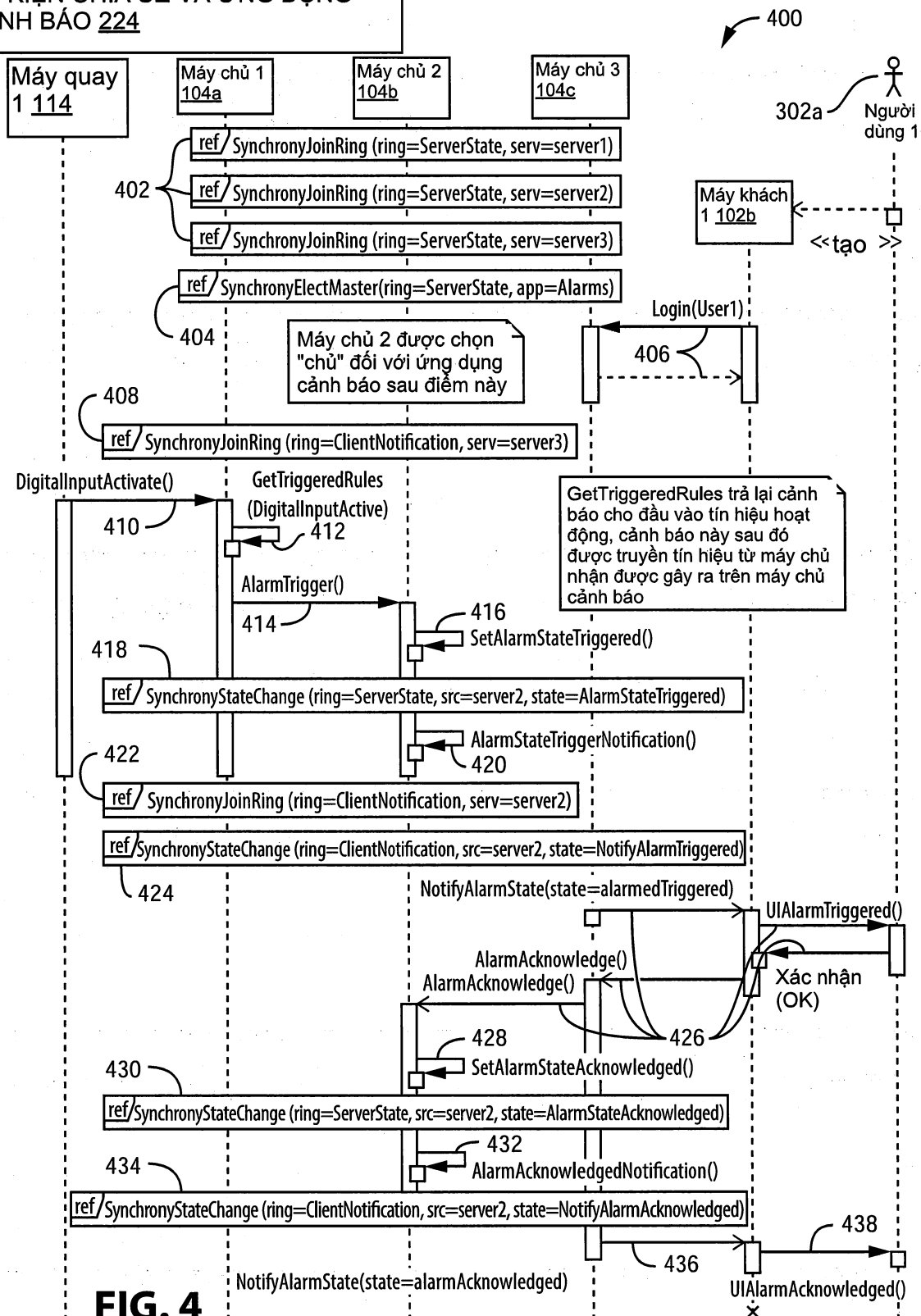


FIG. 2



SỰ KIỆN CHIA SẺ VÀ ỨNG DỤNG
CẢNH BÁO 224



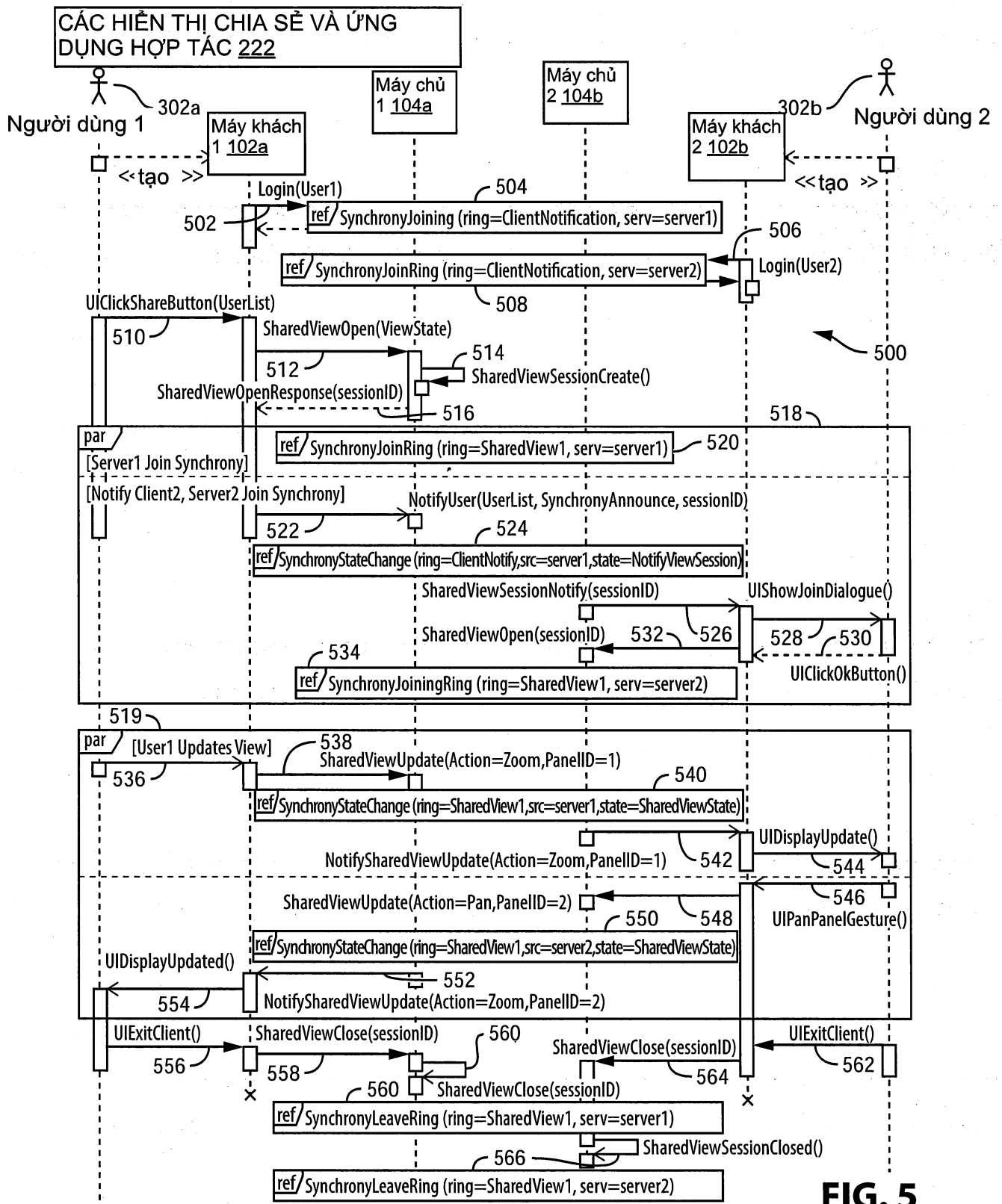


FIG. 5

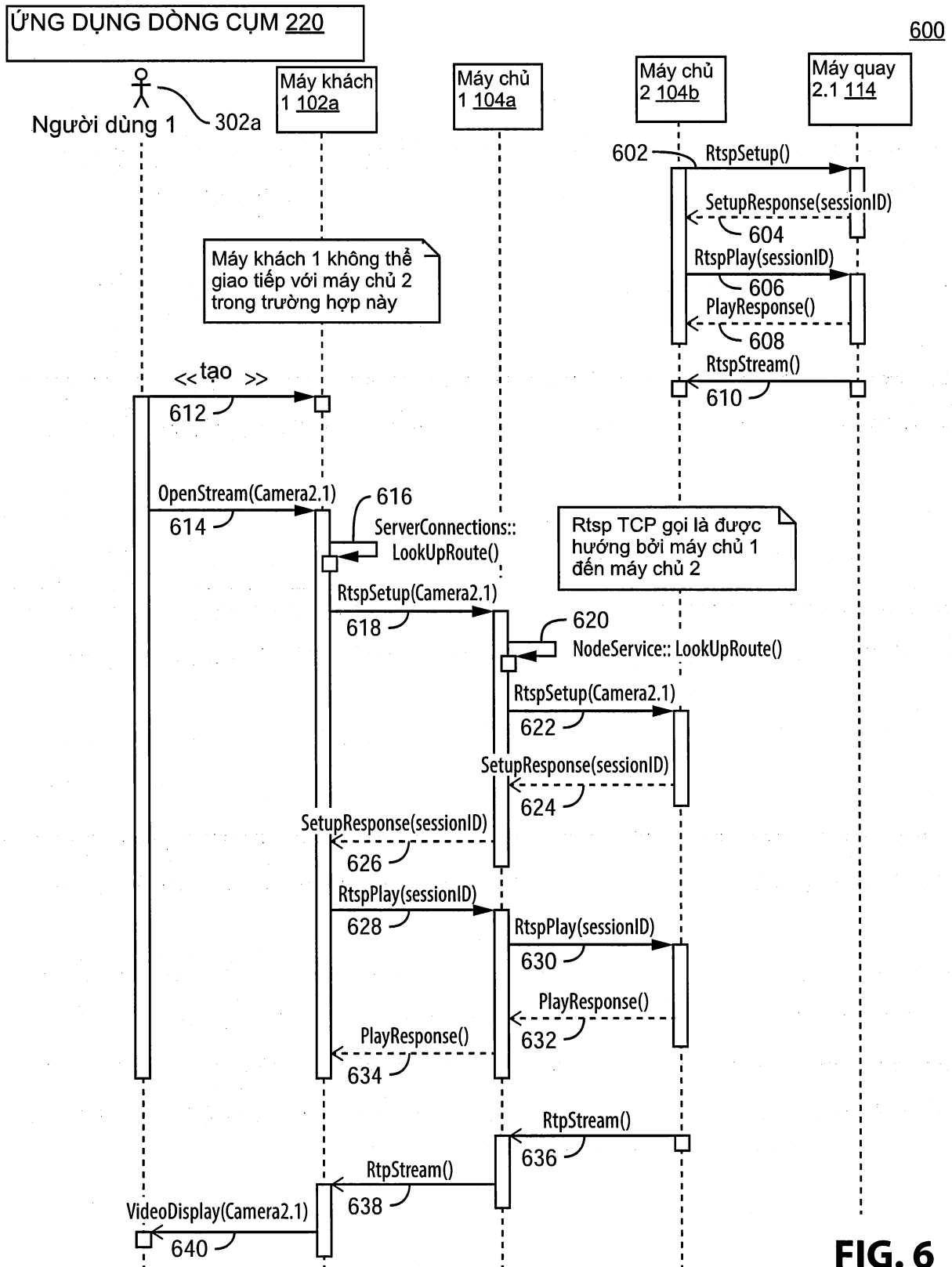


FIG. 6

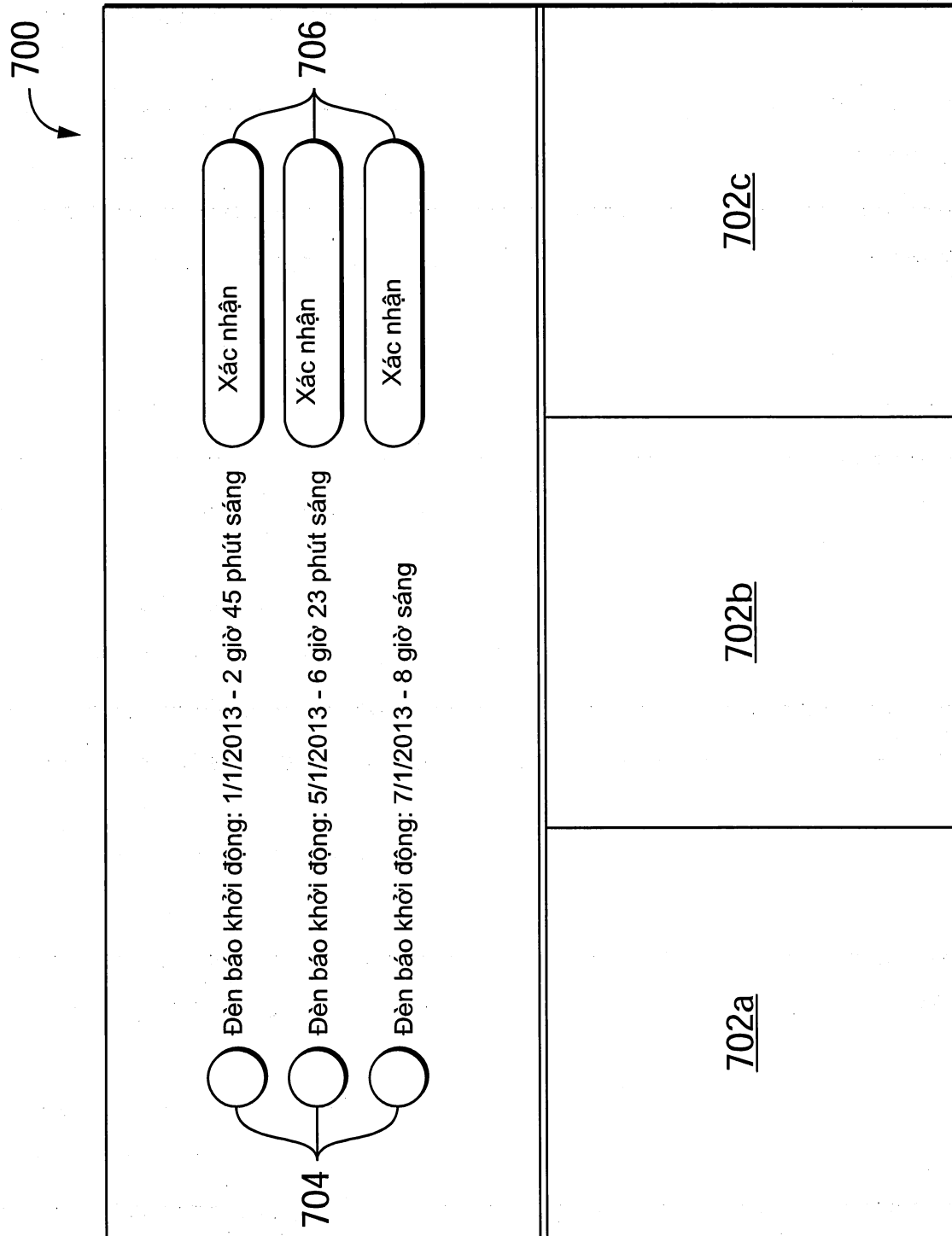
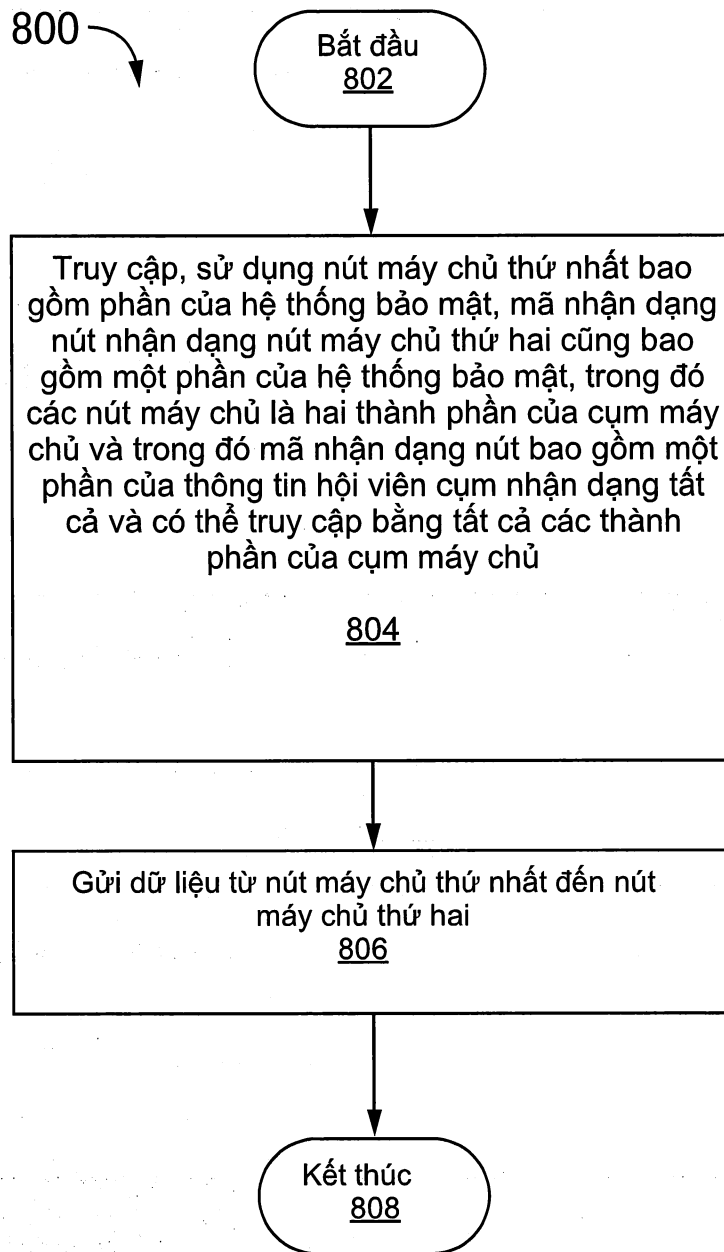
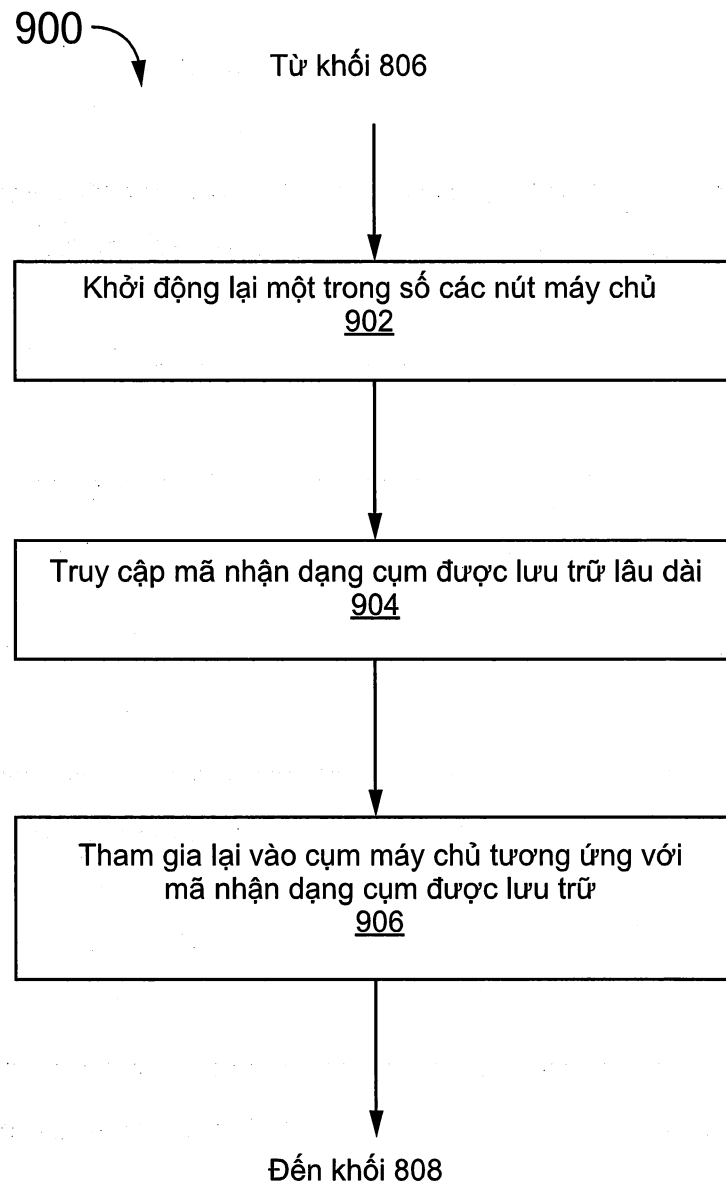
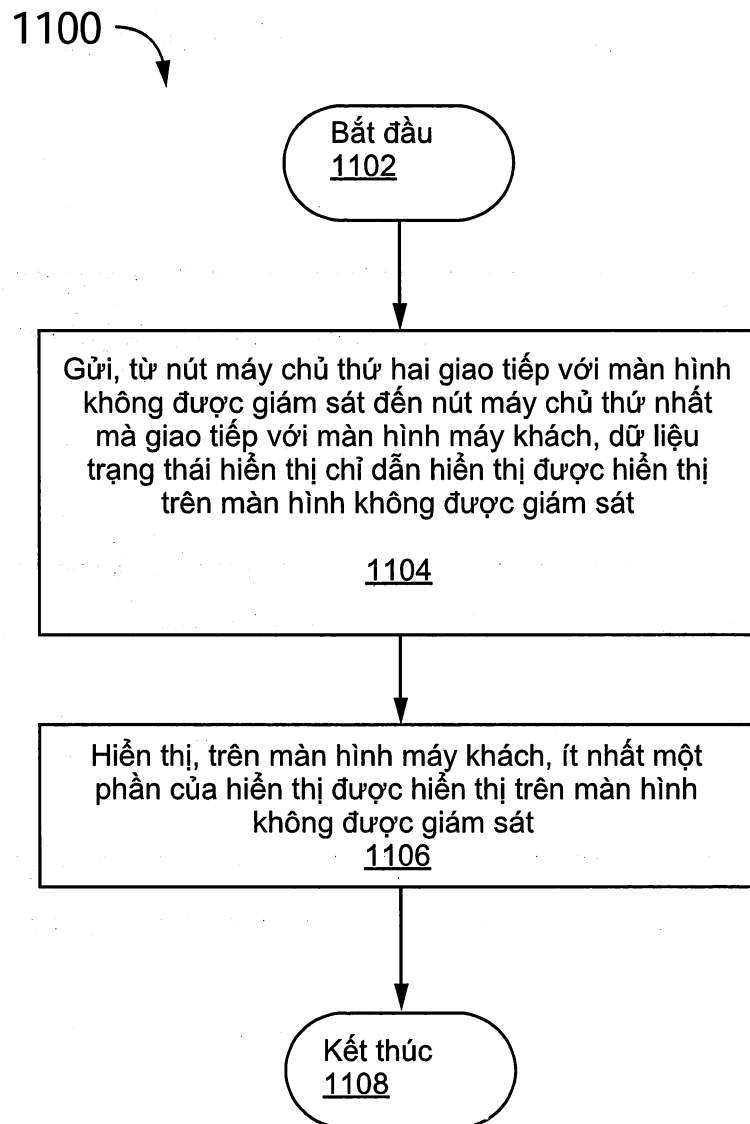


FIG. 7

**FIG. 8**

**FIG. 9**



**FIG. 11**

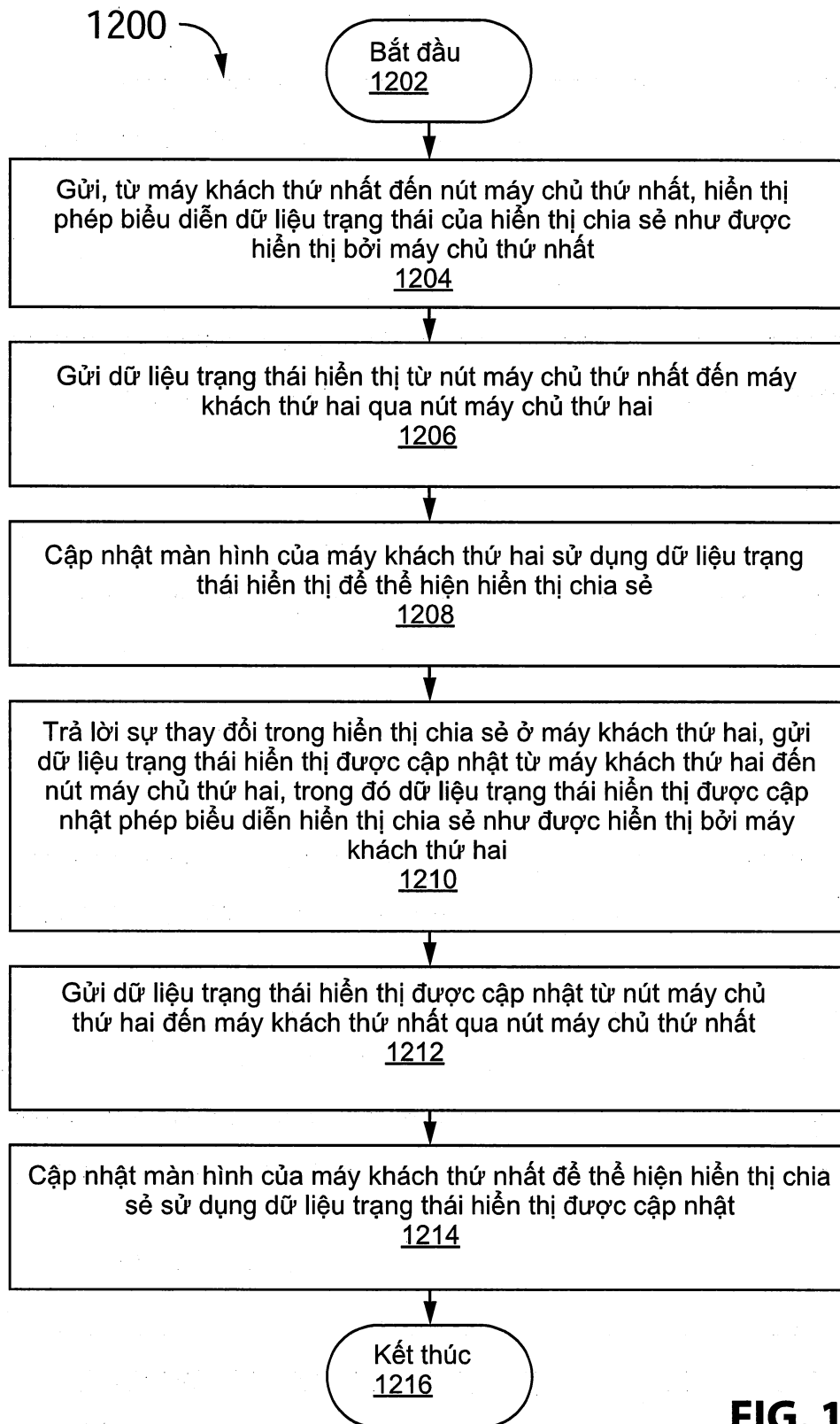


FIG. 12