April 2022

AGENTLESS QUARANTINE OF NETWORK ENDPOINTS

Rajesh Tarakkad Venkateswaran

Veena Ramamoorthy

Saravanan Radhakrishnan

AGENTLESS QUARANTINE OF NETWORK ENDPOINTS

AUTHORS:
Rajesh Tarakkad Venkateswaran
Veena Ramamoorthy
Saravanan Radhakrishnan

ABSTRACT

Techniques are presented herein that enhance a user defined network (UDN) micro-segmentation approach to enable communication between only a select set of endpoints. Aspects of the presented techniques can also restrict the set of endpoints that can initiate communication, thereby enabling specific use cases such as quarantine and remediation workflows, without the need for installing software agents on endpoints. Among other things, aspects of the presented techniques support the dynamic creation of UDN rooms and groups of rooms, do not require configuration changes on intermediate devices, allow administrators to belong to multiple rooms or groups of rooms, and support an audit capability (that helps to detect, inspect, and monitor traffic patterns for endpoints that are under quarantine).

DETAILED DESCRIPTION

It is well known that all of the endpoints in a network are exposed to a very wide range of threats from the services that they access. Once infected, an endpoint poses an even greater threat to the other endpoints in the network for they can now act from within the network. While some of the endpoints may run enterprise monitoring software that constantly monitors the security posture, as well as quarantines infected endpoints, not all of the endpoints can be assumed to be running such software. Further, while it is important to isolate endpoints with a poor security posture, it is just as important that they continue to be accessible to information technology (IT) administrators so that those administrators can remediate the problem.

Effective endpoint security management thus involves: (1) the ability to identify endpoints with a poorer than expected security posture; (2) the ability to classify an endpoint's level of security compromise; (3) the ability to quarantine and limit connectivity to an endpoint, as appropriate, given its compromised security posture; and (4) the ability

to restore normal connectivity once any security issues are remediated. Techniques are presented herein that provide for the ability to the ability to quarantine and limit connectivity to an endpoint, as appropriate, given its compromised security posture and also provide for the ability to restore normal connectivity once any security issues are remediated.

Generally, a user defined network (UDN) is a solution that provides for preventing broadcast traffic from crossing user defined network boundaries. A UDN supports the ability to limit the set of endpoints who can reach each other over unicast or multicast transmissions based on their source and destination media access control (MAC) addresses. Endpoints that should see each other's broadcast traffic may be grouped into a UDN "room."

Presented herein are techniques through which the quarantine of rogue endpoints can be achieved by building upon the capabilities that are available on a UDN. In particular, aspects of the presented techniques employ a Quarantine Engine cloud service that authorized enterprise administrators may employ to group a set of MAC addresses into a UDN room. Such a cloud service may expose authenticated, authorized representational state transfer (REST) endpoints that may be invoked either from an administration application that is running on a mobile application or within an enterprise.

Further aspects of the presented techniques facilitate endpoints that are in one UDN room communicating with endpoints that are in a selected set of other rooms, but not necessarily all of the other rooms. Such a capability is not possible with the current network equipment vendor UDN solutions.

Consider an example sequence of interactions that may take place before the quarantine of affected endpoints is achieved. For example, a UDN room may be created with the MAC address of an affected endpoint when a Detection Engine (DE) identifies that a particular endpoint is vulnerable and/or infected. Such an identification may be done through, for example, a call to an authenticated REST endpoint.

After an endpoint is moved to a different room, an administrator may add a set of administration endpoints, representing the MAC addresses of IT experts who can remediate the problem, into a new privileged room. The room that was created for the affected endpoint and the administration privileged endpoints may now be assigned to a UDN group.

A UDN group represents a set of rooms that can communicate with each other. Importantly, unicast traffic can only be initiated from the endpoints that are in privileged rooms to endpoints that are in privileged or non-privileged rooms.

By following the above approach, only the affected endpoint and the IT devices that are assigned to work with the endpoint would be able to communicate with each other. Such an arrangement prevents the endpoint from reaching, and hence infecting, any other host.

Once the endpoint is remediated, it may be removed from the above-described group and UDN room to restore normal connectivity. Figure 1, below, illustrates elements of an exemplary arrangement according to aspects of the techniques presented herein and reflective of the above discussion.

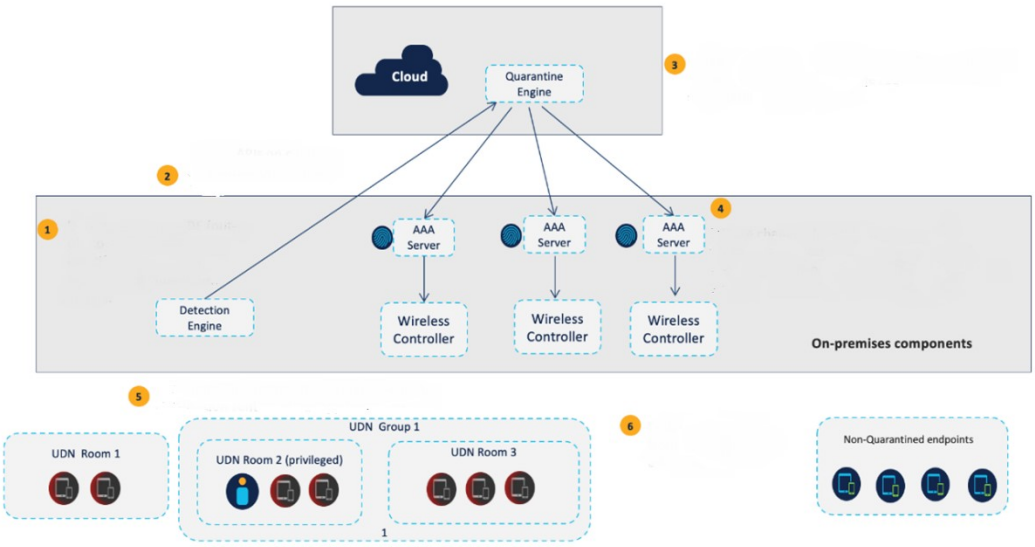

Figure 1: Exemplary Arrangement

Consider a series of steps, which are labeled 1 through 6 in Figure 1, above, as follows. During Step 1, a DE identifies the affected endpoints to a Quarantine Engine. Step 2 encompasses the invocation of application programming interfaces (APIs) on a cloud to quarantine one or more endpoints. Under Step 3, based on the tenant and MAC address the Quarantine Engine sends a room change request to the corresponding authentication, authorization, and accounting (AAA) server.

Step 4 encompasses the initiation of a change of authorization request for the MAC address and the moving of the affected endpoint to a dynamically created UDN room with the appropriate set of IT administration endpoints. Under Step 5 the endpoints are quarantined for corrective action and a notification of same is sent to an administration application for action. Finally, at Step 6, the endpoints are removed from quarantine following the administrative action.

Aspects of the presented techniques encompass a cloud-based Quarantine Engine that is authenticated through the Active Directory of an enterprise that exposes REST-based APIs for performing the actions that are described below.

First, an affected endpoint is assigned to an UDN room. Second, a set of UDN rooms are placed into an UDN group. Endpoints that belong to the UDN rooms in a group may talk to each other. Each group is identified by a group identifier (ID) (i.e., a groupID) that is akin to a UDN ID that identifies a UDN room. Typically, a UDN group will have one privileged room and one or more UDN rooms which contain affected endpoints. Third, a room may be marked as being privileged. Privileged rooms can initiate communication to the other rooms that are in a group while non-privileged rooms cannot initiate such communication.

Aspects of the presented techniques further encompass enhancements to a wireless controller to permit the traffic that flows between endpoints to include a check for the groupID of the source and destination MAC addresses along with the privilege level of the source MAC address.

Use of the techniques presented herein offers a number of advantages over existing solutions. First, no agent is required. The presented techniques do not require any agent to be running on an endpoint. All of the other existing solutions require that an endpoint run proprietary agent software that monitors and enables quarantining. In contrast, the presented techniques only require the MAC address of an endpoint for effective quarantine.

Second, UDN rooms may be dynamically created. Under the presented techniques, UDN rooms may be created on demand with a set of MAC addresses for which communication should be allowed. IT administrators can enable communication with an endpoint only when it is required and they may revoke such permission after they are done. The presented techniques provide a scalable and extremely easy means for creating a

dynamic group of endpoints when compared to traditional virtual local area network (VLAN)-based approaches. Further, no configuration is required on intermediate devices. A Remote Authentication Dial-In User Service (RADIUS) change of authorization request that includes a vendor-specific attribute is sent to the wireless controller to indicate the UDN room and group that a MAC address belongs to. This enables the forwarding of traffic based on the above described rules.

Third, administrators may belong to multiple groups. Under the presented techniques, the same administration endpoint can now interact with multiple endpoints in different UDN rooms. This enables a micro-segmentation of the network based on complex networking needs.

Fourth, an audit capability is supported. Using UDNs to group endpoints, as supported by the presented techniques, benefits from the detailed auditing logs that are available to track the communication between endpoints. This can be of immense help during the detection, inspection, and monitoring of traffic patterns for endpoints that are under quarantine.

Both the Android and the iOS operating systems are implementing changes in their more recent releases to support better privacy through a Randomized and Changing MAC Addresses (RCMA) paradigm.

For example, beginning with the fifteenth major release of the iOS operating system (iOS 15) if a device has not joined a network in the last six weeks then it will employ a different private address the next time that it connects to that network. If a device is forced to forget a network, then it will also forget the private address that it used with that network unless it has been less than two weeks since the last time the device was made to forget that network.

Further, the Android operating system also randomizes a MAC address. However, it uses a persistent randomization unless it has been specifically instructed otherwise by an application. Essentially, the randomized MAC is used at all times and does not often change.

The types of changes that were described above present challenges not just for the techniques presented herein but for all MAC-based authentication solutions.

For the quarantine use cases such as were described and illustrated in the above narrative (according to the techniques presented herein) it is extremely unlikely that endpoints will stay away for upwards of two weeks after being assigned to a group. Additionally, solutions like UDN are moving towards assigning a public key infrastructure (PKI)-based identifier to endpoints rather than relying only on their MAC addresses. It is important to note that the techniques presented herein may in the future also move to using one of these mechanisms.

In summary, techniques have been presented herein that enhance a UDN micro-segmentation approach to enable communication between only a select set of endpoints. Aspects of the presented techniques can also restrict the set of endpoints that can initiate communication, thereby enabling specific use cases such as quarantine and remediation workflows, without the need for installing software agents on endpoints. Among other things, aspects of the presented techniques support the dynamic creation of UDN rooms and groups of rooms, do not require configuration changes on intermediate devices, allow administrators to belong to multiple rooms or groups of rooms, and support an audit capability (that helps to detect, inspect, and monitor traffic patterns for endpoints that are under quarantine).