March 2022

AUTHENTICATED AND SECURE SERVICE DISCOVERY USING HOLOCHAIN

Niranjan M M

AUTHENTICATED AND SECURE SERVICE DISCOVERY USING HOLOCHAIN

AUTHOR:
Niranjan M M

ABSTRACT

mDNS is predominant and lightweight protocol that is used to do service announcements/advertisements and discovery by the enterprise devices. mDNS is by definition multicast in nature and is UDP based non encrypted service announcement and discovery protocol. There are many security challenges for using mDNS as network service discovery protocol in enterprise IoT deployments. One such challenge is to avoid the MITM attacks (network service spoofing, forge the identity etc.,). Today there is neither infrastructure nor method built into the protocol to address the above limitations. Ideally the network should be intelligent to handle this scenario using the sophisticated methods. The techniques presented herein is to provide authenticated and secure Service Discovery in an enterprise deployment using the Holochain technology and provide secure service registration, service discovery, service transfer, service auditing and usage.

DETAILED DESCRIPTION

Service advertisement and discovery is the common practice to crawl the networked services available in an enterprise (wired/wireless) network deployment. mDNS (multicast DNS) is one such protocol predominantly used in enterprise IoT to announce/discover the services, as it requires little or no administration or configuration to set the service advertising network up and running.

mDNS is a UDP based and insecure (neither encryption nor authenticated registration) protocol and can be spoofed by anyone to mimic as service owner, this can be dangerous in case of enterprise IoT deployments wherein on-boarding user (BYOD devices) will be relying on the discovered service details from mDNS controller/network element as authentic.

mDNS is prone to MITM (Man-In-The-Middle) attack due its nature of protocol operation and limitations. Some of the attacks include,

- mDNS Poisoning Attacks
- Spoofing Attacks and forge the Identity

Today there is neither infrastructure nor method built into the protocol to address the above limitations. Ideally the network should be intelligent to handle this scenario using the sophisticated methods.

The techniques presented herein is to provide authenticated and secure Service Discovery in an enterprise IoT deployment using the Holochain technology and provide secure service registration, service discovery, service transfer, service auditing and usage. It addresses the MITM attacks (viz., network service spoofing etc.,) and helps in service roaming, device replacement in an authenticated and secure way.

The techniques presented herein are explained in below steps to address the limitations of mDNS by having authenticated and secure service discovery using Holochain:

Device (in-turn Service) Authentication:

1. Network elements (Switches/Routers/WLCs) in the enterprise deployment are registered to the Holochain network as Trusted participants.
2. Device (wired/wireless) which announces network service (example, printer as _ipp._tcp.local) is deployed on to the network.
3. The network element (Switch/Router/WLC) authenticates the device using dot1x/MAB/WebAuth with the identities provided (exchanged) during this process.
4. The Identities include MAC, IP, username, serial number, device identity, device make, device model etc.,
5. After successful authentication, the network element will add the device with its identities along with service type (Printer, appleTV, etc.,) to the local hash-chain.
6. Now device providing the service is registered with the network element.
7. Network element would publish the registered device along with its identities to the Holochain DHT as valid network service provisioning device along with the service type. Each entry in the local hash-chain and the shared DHT will have the identities of the authenticated device and the service that is owned by that device.

For Example:

- Published device with device identity ID_X entry would be linked to service name "X".
    - Ex: Printer located in BUILDING-18, 3rd floor with service name "bgl18_floor3_print1_ipp._tcp.local" would be published to the Holochain DHT and this published entry will be linked to its identity with MAC address as "aabb.ccdd.eefa" or IP address as "192.168.1.2".
- Anyone who know service name X can find the device ID_X entry in the DHT.
- With the concept of "anchor" in Holochain DHT, the service name "X" would also be linked to "_all_services_" anchor. This anchor should be already existing in the DHT, and its value can be hard-coded or created manually in the holochain app (running on the Trusted participants).
    - Ex: Printer service with service name "bgl18_floor3_print1_ipp._tcp.local" will be linked to "_all_printers_" anchor.
- Anyone can query the anchor _all_services_ for a full list of services.
    - Ex: Anybody wants to know about all the printers in the enterprise deployment, would query the anchor with name "_all_printers_" and get all the registered printers from the DHT.
- And, to make it flexible, Holochain DHT provides concept of "tag", using that we can group the services based on the tag which is added as metadata into the DHT. This acts like filter to reduce the DHT lookup.
    - Ex: Printers in BUILDING-18, 3rd floor can be given with tag as "BGL18_3_PRINTER", so that we can list only the printers available in the BUILDING-18 3rd floor.

Figure-1 depicts the enterprise deployment to have authenticated and secure service discovery using Holochain.

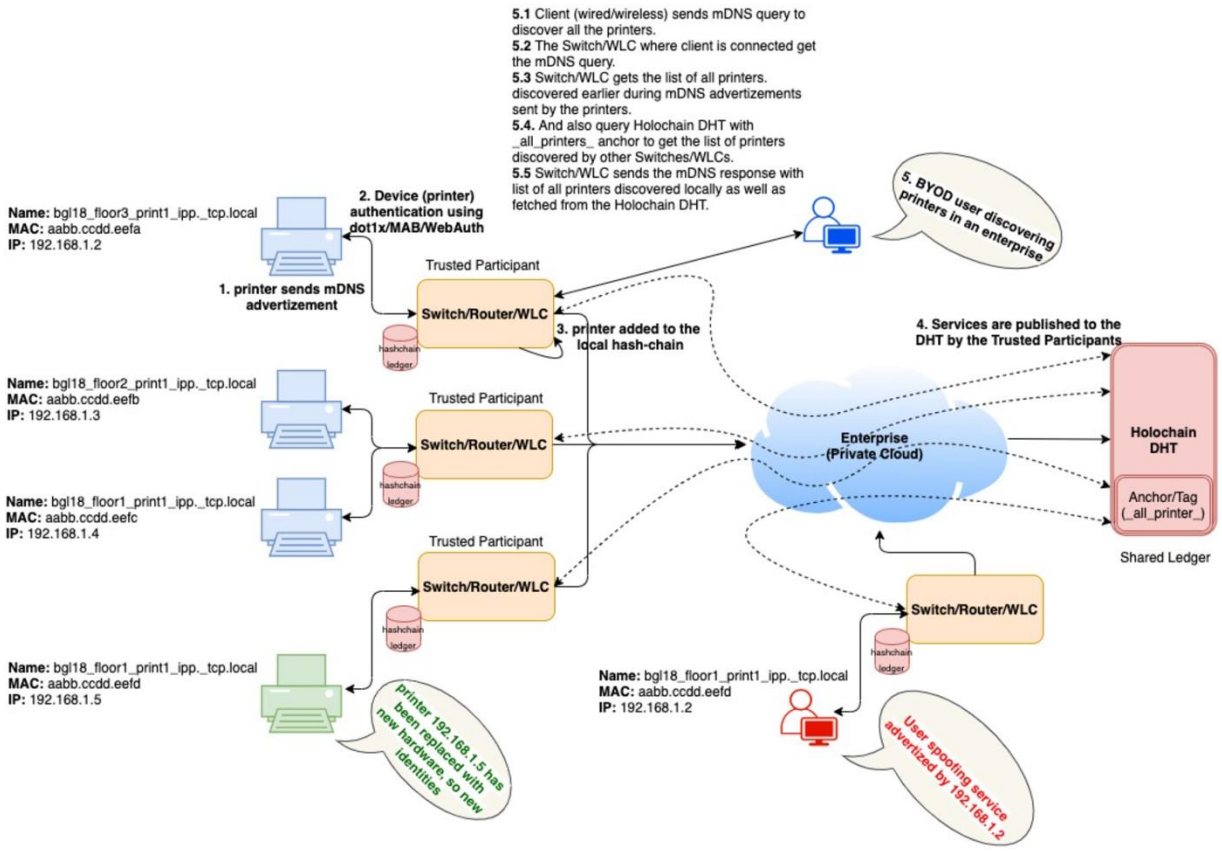

Figure-1

A. How spoofed services are detected:
Note:
- Service name X could be "bgl18_floor3_print1_ipp._tcp.local" with IP address "192.168.1.2".
- Device Identity ID_X could be with MAC address "aabb.ccdd.eefa" (authenticated device/service)
- Spoofed device identity ID_Y could be MAC address "aabb.ccdd.eefd" (spoofed device/service) as shown in figure above.
- network element D1 could be Switch/WLC.

1. Let us say, Service with service name "X" with identity ID_X is authenticated by the network element (Trusted participant) D1 and added to the local hash-chain.
2. Network element D1 publishes Service with service name "X" with identity ID_X to the Holochain DHT.
3. Let us say, Service with service name "X" with different identity ID_Y is trying to advertise on to the network.
    - The Service with service name "X' with identity ID_Y is not found in the local hash-chain of network element D1.
    - Then the network element D1 will look up the DHT for the authenticity of the service with service name "X".

- The DHT lookup to validate the service with service name "X" will fail, so the service registration will be denied.
    - In other words, the DHT lookup for the service with service name "X" and returns the entry which has the original identity ID_X of the device which was added earlier to the DHT.
    - These spoofed events will be cached on each network element and can be propagated across the enterprise network for future evaluations (for past failure analysis for the behavioural validation).

B. How Service Roaming is handled:
    1. Let us say, Service with service name "X" with identity ID_X is roamed across the network from network element D1 to network element D2.
    2. The device with identity ID_X will be re-authenticated against D2 and added to the local hash-chain of D2.
    3. D2 publish Service with service name "X" with identity ID_X to the Holochain DHT. In other words, it propagates the information about Service with service name "X" with identity ID_X is visible at DHT.
    4. Holochain DHT validates the hash for Service with service name "X" with identity ID_X as registered by the D1 earlier with the hash for Service with service name "X" with same identity ID_X registered now by D2.
    5. Since authenticated Service with service name "X" is earlier registered by D1 and now re-registered by D2, DHT consider this as roam event and update the service roam transaction to the DHT. Now onwards, DHT returns any query for the service with service name "X" from the entry of network element D2.

C. How Service Transfer is done:
    1. If the service with service name "X" with identity ID_X was attached to the network element D1 is replaced with the device with identity ID_Y and attached to the same network element D1.
    2. Network element D1 authenticate the device with new identity ID_Y associated with service with service name "X".
    3. Network element D1 added the service with service name "X" with the identity ID_Y to the local hash-chain.
    4. As we know, Holochain provides privileges under Validation Rules for Happ running on network element D1 to create/update/remove the entry from the DHT.
    5. As same service with service name "X" is associated with the identity ID_Y, network element D1 (which is a trusted participant) will update the entry by mapping service with service name "X" with the identity ID_Y to the DHT. Now onwards, DHT returns any query for the service with service name "X" returns an entry with identity ID_Y.

The techniques presented herein provides authenticated and efficient Service Discovery in enterprise deployments. It also provides secure service registration, service roaming and service transfer in IoT enterprise network deployments. Moreover, this method can be used in enterprise wireless/wired deployment for discovering services such as Printers, AppleTV etc., In general, this method can be used in any enterprise deployments where services are advertised and discovered using mDNS protocol.