Errata - 2.1 Updating MAC Addresses from Nmap

The Issue(s)

Up to and including 2.1 we have an issue where we were not populating, nor updating the IP table with the MAC Address retrieved from the Nmap scan (if any) during Discovery.

This ONLY affects devices that are only discovered using Nmap. IE, they have no working credentials or don't expose WMI, SSH or SNMP for Open-AudIT to query.

Only devices on the subnet that is local to the Open-AudIT server can have their MAC address retrieved using Nmap - so remote subnet devices are not affected. Those will not return a MAC address anyway.

We only added (we never updated) an IP row when we found a NEW device. We never updated this row on subsequent discovery runs. If a device had an IP row without a MAC, then subsequent discovery runs reported a MAC, we did not update this row (see below Technical Stuff for why). That is besides the fact we weren't parsing the MAC address from the discovery script anyway.

The Files

The attached files are included in the 2.1.1 release, but are provided here for users who are adversely affected.

There are two files worth of fixes. The relevant paths are -

Windows

```
c:\xampplite\open-audit\code_igniter\application\controllers\include_input_discoveries.php

c:\xampplite\open-audit\code_igniter\application\models\m_devices_components.php
```

Linux

```
/usr/local/open-audit/code_igniter/application/controllers/include_input_discoveries.php

/usr/local/open-audit/code_igniter/application/models/m_devices_components.php
```

You should first backup your existing files (just rename them .php.bak), then copy the attached to those places on your filesystem.

Rerun a discovery which has a device you know has not had its MAC Address populated and see if you get the expected result.

include_input_discoveries.php

m_devices_components.php

Technical Stuff

#1 - Not populating the MAC address from the discovery script is a bug, pure and simple. My apologies 😞

#2 - Only adding an IP entry when we find a NEW device (ie, never updating an IP on subsequent discovery runs with a MAC or any other attributes), was a choice made at the time. We cannot use the existing "update components" routine. That would remove other valid existing IPs from that device.

An example -

We have a Windows device with several IP addresses (maybe several network cards, maybe multiple IPs on a single card, doesn't matter), populated from an audit script.

On a subsequent discovery run, if we used the existing "update components" routine, we would only have a SINGLE ip address (at the point this code is called). This occurs ***before*** we run the audit script. The other non-matching IPs would therefore be marked current = 'n' and change logs generated. This is obviously incorrect.

This was a known design issue at the time, but was thought to not be that important because we would populate the IPs once we had run the audit script. Obviously devices where we cannot run an audit script break this logic.

The Fix

I have created an additional function to be used ONLY by the discovery routine to process and update an IP entry. The function is in m_devices_components and is called nmap_ip. This is ONLY called by the discovery processing code. The logic is below.

Check if we have a matching IP and both discovered and database MAC match (including if empty) - UPDATE.

Check if we have a matching IP, no discovered MAC, but a MAC in the database - UPDATE (will not overwrite existing database MAC in the IP record).

Check if we have a matching IP, a discovered MAC, but no MAC in the database - UPDATE (will add the MAC to the existing database IP record).

If none of the above work, insert a new IP entry.