

Ok,

Phase 1 Complete: You now have what I have at the office. Here are some key points/todo's:

- 1) Entering Device Types will not affect scanning. There is an incomplete function called `mactrack_scanner.php` that requires modifications prior to you being able to add/remove devices from the Device Types table and have that impact scanning.
- 2) There are spurious UI issues. Some have been reported to me, but I have yet been able to make time to fix. I use phpMyAdmin to add/remove rows from the main tables right now.
- 3) There are a few issues with the proper detection of trunk ports on Cisco devices. This will only impact scanning if some "did I do that" local configuration corner cases.
- 4) The "ignore ports" field is currently non-functional. I will get this fixed pretty quickly. This only has an impact if you are using a user port as an uplink (cascading switches).
- 5) The reset of the database maintenance time under Settings is not 100% functional. Again, time is (not in this case) money, and I have not made a cent with this...

Customization notes:

The method that I have chosen to implement vendor specific scan's is to create a file under `/lib/mactrack` called `"vendor_functions.php"` where "vendor" is your vendor. For example, I have `"mactrack_cabletron.php"` and `"mactrack_cisco.php"` today. I would expect an `"mactrack_hp.php"` and others moving forward.

The core functions are included in `"mactrack_functions.php"`. These functions include:

**<b>valid\_snmp\_device</b>** - this function will take a host and first use the `snmp_readstring` specified for the host. If that is a valid readstring, it will return TRUE. If not, it will scan the list of `snmp_readstrings` in the host, and if it finds a good one, it will use it instead and update the host with the new restrng as the default. If none of the readstrings work out, the host is marked "down".

**<b>find\_scanning\_function</b>** - this is an incomplete function that will utilize the host information and attempt to locate the correct scanning function and any special circumstances around that function.

**<b>get\_arp\_ip\_ranges</b>** - this function scans the arp table of a device and inserts the ordered mac/ip pairs into a temporary table for later association with the device mac's.

**<b>get\_generic\_switch\_ports</b>** - this will likely be a popular scanning function for an otherwise dumb switch. It will work with any switch where VLAN information is not required. You pass it a few standard parameters and off it goes.

**<b>get\_base\_dot1dTpFdbEntry\_ports</b>** - this is a core function that scans the dot1d bridge table and returns valid ports to the parent function for subsequent processing. This function is critical to the overall scanning process.

**<b>xform\_mac\_address</b>** - this function will take an ugly mac address from snmp and format it as such "XX:XX:XX:XX:XX:XX". Pretty boring.

**<b>xform\_standard\_indexed\_data</b>** - this function takes an OID, a device and an alternate readstring as input and returns the last octet of the OID as an index and the value as a value pair. This is a very common translation function throughout.

**<b>xform\_cisco\_workgroup\_port\_data</b>** - this function is similar to the xform\_standard\_indexed\_data function with the exception that Cisco, for reasons unknown to me, decided to make the last 2 octets mean something like card/port. Therefore, it uses the last two to build the relationship rather than the last octet.

**<b>xform\_indexed\_data</b>** - this function combines the features of the previous 2 functions in that you can specify the number of octets to use in determining the association. The function could, although I haven't gotten around to it, replace the two prior functions.

**<b>db\_process\_add</b>** - this function essentially adds a process entry to a table that keeps track of the number of concurrently running data collection processes.

**<b>db\_process\_remove</b>** - this function removes a process from the process table.

**<b>db\_update\_device\_status</b>** - this function updates a device, once it has been scanned using the mactrack\_scanner.php function. It will update port counts, snmp readstrings, etc.

**<b>db\_store\_device\_port\_results</b>** - this function is pretty straightforward, once you have made the association of MAC to port for the entire switch, this function stores that data into a temporary table.

**<b>mactrack\_walk</b>** - this function is essentially no longer required. It at one time optimized the snmpwalk process with snmpv2 devices by using the bulkwalk process. Since I have added this to the base 0.8.6h product, I will, at some time, remove the function.