MAC Address Spoofing: a Threat Assessment Approach

Krystof Litomisky

Fall 2011

Although MAC addresses are designed to be globally unique, this is not always the case due to issues such as counterfeiting or manufacturing errors. In this project, I calculate the probability of seeing a MAC address conflict under various conditions. I use real-world data for the number of network devices shipped globally to estimate the total number of network devices in use, and estimate the probability of seeing a conflict for different rates of counterfeiting as well as different sample sizes.

1. Introduction

Under the IEEE 802 standard, each network interface device (including network cards in computers, portable WiFi-enabled devices, as well as individual ports in bridges or routers) should have a globally unique 48 bit MAC address. Blocks of addresses should be purchased by an organization to be assigned to the devices that organization manufactures. However, in some cases a device may not have a globally-unique MAC address. Some reasons for this include:

- Counterfeiters do not pay for MAC addresses
- Low-cost network-card manufacturers may be assigning their cards randomly generated MAC addresses
- Users or system administrators may temporarily assign their devices arbitrary MAC addresses through software.

For these reasons, two or more devices may have the same MAC address. This becomes a problem if the devices with the same MAC address are on the same broadcast domain (e.g. layer 2 network). If the devices are not on the same broadcast domain, then no issues arise, since MAC addresses are not forwarded through routers in layer 3 of the OSI model.

This project estimates the required number of total MAC addresses seen (i.e. sample size) necessary to detect MAC address conflicts or fraudulent MAC addresses for three distinct situations:

- A network interface device supplier steals an unallocated block of MAC addresses from the IEEE
- Devices on locally-administered broadcast domains generate their own random MAC addresses
- Counterfeiters use MAC addresses they have not purchased

This report should be viewed in color.

2. Background

MAC addresses are 48 bits (6 bytes) long. Blocks of $2^{24} = 16{,}777{,}216$ addresses are sold to organizations by the IEEE. Each address in such a block has the same first three bytes, known as the

Organizationally Unique Identifier (OUI). Each organization can distribute the addresses within its OUI block in any manner it pleases, as long as each device gets a unique address [3].

2.1 Total number of network devices

According to the IEEE OUI Public Listing [2], 15,666 distinct OUIs were assigned as of November 9th, 2011. This represents a potential $15{,}666 \times 2^{24} = 262{,}831{,}865{,}856$ unique, valid MAC addresses that could be in circulation in the world. Clearly, however, not all of these devices are in use. The following table shows the values I used for the numbers of network devices currently in use:

	Manufactured (mil)	year	source
PCs	345	2010	[11]
Smartphones	478	2011	[10]
Internet-Enabled Consumer Devices	161	2010	[9]
Routers	69	2010	estimated
Total manufactured per year	1053		
Average lifetime: 4 years			
Total devices in operation	**4,212**	**million**	

Table 1: Estimated total number of network devices in operation

I was unable to find information regarding the number of network routers shipped or in use, and therefore approximate the total number of routers in use: I assume that there are 5 PCs shipped for every router. Thus, I estimate the total number of routers shipped in 2010 to be $\frac{345}{5} = 69$ million.

I further assume that a typical network-enabled device has an average lifetime of 4 years. This leads to a final figure of 4.21 billion network devices in use globally.

2.2 Techniques

I calculated various probabilities of interest with different parameters in MATLAB. Where possible, I used figures based on real-world market data, and considered worst-case scenarios for various assumptions.

3. Supplier steals a block of unallocated addresses

This section is concerned with suppliers stealing blocks of unallocated MAC addresses from the IEEE. My goal here is to calculate the probability of seeing an address that has not been properly purchased from the IEEE. Hereafter the range of addresses that have not been assigned by the IEEE to any manufacturer is referred to as the "unassigned range".

Let N be the total number of MAC addresses assigned to devices and currently in use.

Let v be total number of these addresses that were properly purchased from the IEEE, $v < N$.

Let n be the sample size, $n \leq N$.

Let p be the probability of seeing an address that has not been properly purchased from the IEEE. I calculate p by first calculating $\bar{p} = 1 - p$, the probability of seeing no addresses from this range.

$$\bar{p} = \frac{v}{N}\frac{(v-1)}{N}\frac{(v-2)}{N}\ldots\frac{(v-n+1)}{N}$$
$$= \frac{v!}{N^n(v-n)!}$$

Then $p = 1 - \bar{p} = 1 - \frac{v!}{N^n(v-n)!}$.

3.1 Results

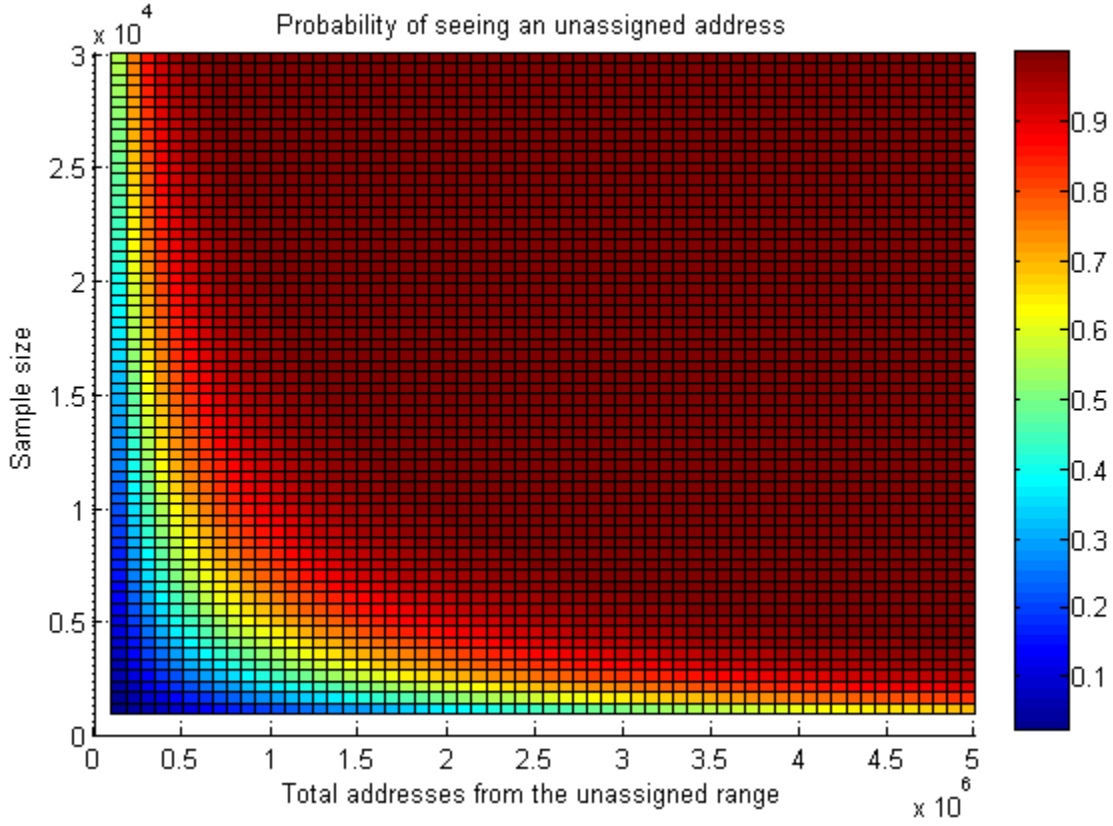

Figure 1: probability of seeing an address from the unassigned range

Figure 1 shows the probability of seeing an address from the unassigned range. Note a few interesting values:

- If 5 million devices in the world had a MAC address from the unassigned range, a sample size of approximately 2000 is necessary to have a probability of 90% of detecting at least one such address. To have 99% probability, a sample size of 3,900 is required.
- If 1 million devices in the world had a MAC address from the unassigned range, a sample size of 9,700 is required to have a probability of 90% of detecting at least one such address. To have 99% probability, a sample size of 19,367 is required.
- If only 100,000 devices in the world had a MAC address from the unassigned range, a sample size of 30,000 would yield a probability of only 56% of detecting at least one such address.

4. Randomly-generated MAC addresses for locally-administered space

In addition to the possibility of fraudulent MAC addresses being assigned to devices by counterfeiters or low-cost manufacturers, there are also ways to change the MAC address that is reported by the operating system for network devices. The authors of WhatIsMyIPAddress.com cite several reasons for why someone might want to change their MAC address [6]:

- Privacy concerns
- To be able to use multiple machines with internet service providers who bind their service to specific MAC addresses
- To get around MAC-address-tied software licenses

I have found many sources of information on how to change the MAC address that is reported by the operating system. In most of these, the decision of what the MAC address should be set to is left entirely to the user. In other words, such addresses are not generated randomly. In a worst case scenario, users would simply use whatever address was used in the tutorial they followed to change their MAC address. Thus, certain addresses would be picked exceedingly often.

However, a separate search for “generate random MAC address” found additional solutions, including instructions for generating random MAC addresses. SMAC seems to be a popular tool for MAC address spoofing under Windows [7]. For *nix and Apple OS X, changing the MAC address via the command line seems to be the primary method of choice. Since I do not know how the authors of SMAC generate their addresses, I will focus on the command line methods.

There is a number of different methods to generate random MAC addresses via the command line; for instance, the MD5 hash function can be used with the amount of time the computer has been running as follows [8]:

```
$(md5sum /proc/uptime | sed 's/\(..\)/\1:/g' | cut -c1-17)
```

The above command (and others) depends on the time the computer had been running; therefore, since MD5 is widely accepted as a good hash function, the MAC addresses generated in this way will have a good distribution as long as addresses are not generated in a startup script.

To better understand how likely it is that generating MAC addresses locally will cause issues, it would be necessary to do a thorough user survey. This is beyond the scope of this project.

5. MAC address conflicts in valid ranges

This section is concerned with conflicts in MAC addresses from blocks that have been properly purchased from the IEEE. This can be the result of counterfeiters who use addresses with an already-assigned OUI, or manufacturing errors. The case of locally-administered addresses (see section 4) is not covered here.

In this section, I assume that the stolen (or counterfeit, or faulty) addresses are distributed randomly throughout the valid range. This is not always true: there are numerous reports on the web of people buying two or more hardware devices of the same type that come with identical MAC addresses (see for

instance [4,5]). However, assuming that the MAC addresses are distributed randomly is a worst-case assumption: conflicts would be easier to detect if the stolen addresses were distributed in a different fashion.

Let N be total number of distinct MAC addresses assigned to devices and in use. In this scenario, this is also the total number of valid MAC addresses assigned and in use, since the stolen MAC addresses are in the same range.

Let s be the total number of stolen MAC addresses that are assigned to devices and in use, $s < N$.

Let n be the sample size, $n \leq N$.

Let n_s be the total number of stolen addresses that appear in the sample, $n_s \leq n$.

Let n_v be the total number of valid addresses that appear in the sample, $n_v = n - n_s$.

Then the probability of seeing no conflict in the sample is given by

$$\begin{aligned} p(\text{see no conflict}) &= p(\text{only valid addresses in } n) + p(\text{valid and stolen in } n, \text{no conflicts}) \\ &= p_1 + p_2 \end{aligned}$$

where $p_1 = p(\text{only valid addresses in sample})$ and $p_2 = p(\text{valid and stolen in sample, no conflicts})$.

Then

$$p_1 = \left(\frac{N}{N+s}\right)\left(\frac{N-1}{N+s-1}\right)\left(\frac{N-2}{N+s-2}\right) \dots \left(\frac{N-(n-1)}{N+s-(n-1)}\right)$$

and

$$p_2 = \sum_{n_s=1}^{n} p(n_s) p(\text{none of the } n_s \text{ stolen addresses in sample match another address in sample} \mid n_s)$$

where

$$p(n_s) = \left(\frac{s}{N+s}\right)\left(\frac{s-1}{N+s-1}\right) \dots \left(\frac{s-(n_s-1)}{N+s-(n_s-1)}\right) \cdot \left(\frac{N}{N+s-n_s}\right)\left(\frac{N-1}{N+s-n_s-1}\right) \dots \left(\frac{N-(n_v-1)}{N+s-n_s-(n_v-1)}\right) \cdot \binom{N}{n_s}$$

and

$$p(\text{none of the } n_s \text{ stolen addresses in sample match another address in sample} \mid n_s) = 1 - [p(\text{a valid matches a stolen}|n_s) + p(\text{a stolen matches a stolen }|n_s)]$$

where

$$\begin{aligned} p(\text{a valid matches a stolen }|n_s) &= 1 - p(\text{no valid matches any stolen}|n_s) \\ &= 1 - \left(\left(\frac{N-1}{N}\right)^{n_s}\right)^{n_v} \\ &= 1 - \left(\frac{N-1}{N}\right)^{n_s n_v} \end{aligned}$$

and

$$\begin{aligned} p(\text{a stolen matches a stolen}|n_s) &= \sum_{i=1}^{n_s-1} \sum_{j=i+1}^{n_s} p\left(\text{conflict between } i^{\text{th}}\text{stolen and } j^{\text{th}}\text{stolen}\right) \\ &= \sum_{i=1}^{n_s-1} \sum_{j=i+1}^{n_s} \frac{1}{N} = \sum_{i=1}^{n_s-1} \frac{n_s - i}{N} \end{aligned}$$

Thus,

$$p(\text{none of the } n_s \text{ stolen addresses in sample match another address in sample} \mid n_s) = 1 - \left(\left(1 - \left(\frac{N-1}{N}\right)^{n_s n_v}\right) + \sum_{i=1}^{n_s-1} \frac{n_s - i}{N}\right)$$

which gives

$$p_2 = \sum_{n_s=1}^{n} \left(\frac{s}{N+s}\right)\left(\frac{s-1}{N+s-1}\right) \dots \left(\frac{s-(n_s-1)}{N+s-(n_s-1)}\right) \cdot \left(\frac{N}{N+s-n_s}\right)\left(\frac{N-1}{N+s-n_s-1}\right) \dots \left(\frac{N-(n_v-1)}{N+s-n_s-(n_v-1)}\right) \cdot \binom{N}{n_s} \cdot \left[1 - \left(\left(1 - \left(\frac{N-1}{N}\right)^{n_s n_v}\right) + \sum_{i=1}^{n_s-1} \frac{n_s - i}{N}\right)\right]$$

Finally, the probability of seeing a MAC address conflict in this scenario is given by

$$p = 1 - p(\text{see no conflict}) = 1 - (p_1 + p_2)$$

5.1 Results

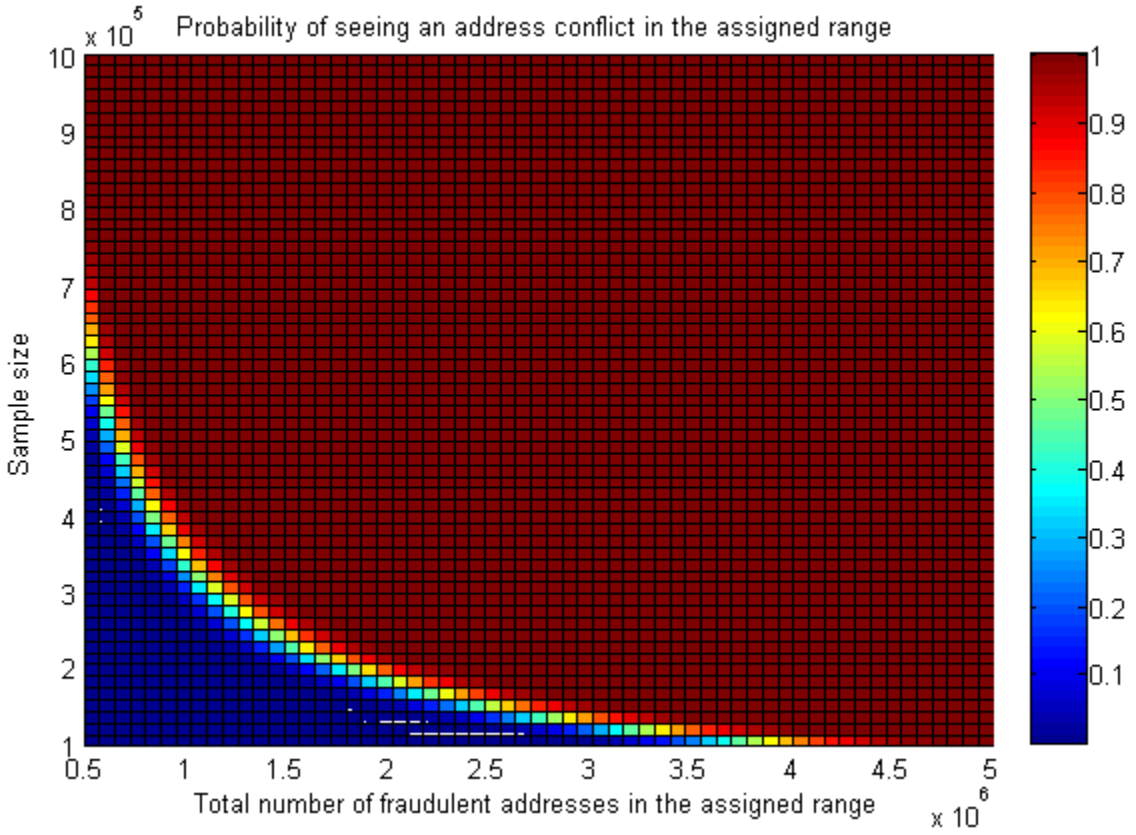

Figure 2: probability of seeing a MAC address conflict in the assigned range

Figure 2 shows the probability of detecting a MAC address conflict in ranges that have been properly purchased from the IEEE (hereafter referred to as the "assigned range"). Note a few interesting results:

- If 500,000 devices in the world fraudulently had an address in the assigned range, a sample size of approximately 685,000 would be necessary to have a 90% probability of seeing a conflict. For 99% probability, a sample size of approximately 760,000 would be necessary.
- For a sample size of 100,000, there would have to be a total of approximately 4,300,000 devices with a fraudulent address in the assigned range to have a 90% probability of detecting a conflict. For 99% probability, there would have to be a total of approximately 4,700,000 devices with a fraudulent address in the assigned range.
- The above observations as well as the plot in figure 2 show that compared with the scenario in section 3, this scenario shows a much more abrupt increase in probability versus sample size as well as total number of fraudulent addresses. For instance, for a given number of total fraudulent addresses in the assigned range, the probability jumps from near 0 to near 1 within a small range of sample sizes.

Conclusions

I have calculated the sample sizes necessary to have sufficient probability of detecting a MAC address conflict in various conditions. The required sample sizes are not prohibitively large: to have a 99% probability of detecting a conflict (for 500,000 devices with fraudulent addresses globally) a sample size of only 760,000 is necessary. It would be interesting to gather a sufficiently large number of MAC addresses to see how many conflicts arise; from this data, the total number of devices with fraudulent MAC addresses in the world could be inferred.

References

[1] CS 204 Fall 2011 Group Project. http://www.cs.ucr.edu/~mart/204/CS_204_Project_11F2.html.

[2] IEEE ISO/IEC 8802 OUI Public Listing. http://standards.ieee.org/develop/regauth/oui/oui.txt. Retrieved November 9, 2011.

[3] Guidelines for use of the 24-bit Organizationally Unique Identifiers (OUI). http://standards.ieee.org/develop/regauth/tut/eui.pdf. Retrieved November 20, 2011.

[4] SlateDroid forums: Duplicate MAC Address. http://www.slatedroid.com/topic/3346-duplicate-mac-address/

[5] Networked Media Jukebox forums: Duplicate MAC address. http://www.networkedmediatank.com/showthread.php?tid=56572

[6] WhatIsMyIPAddress.com: How to change my MAC address? http://whatismyipaddress.com/change-mac

[7] SMAC Official Website. http://www.klcconsulting.net/smac/

[8] m8t's blog: Random MAC Address. http://blog.mmassonnet.info/2007/03/random-mac-address.html

[9] iSuppli: Internet-Enabled Consumer Electronics Devices to Enjoy 50 Percent Shipment Growth in 2011. http://www.isuppli.com/Home-and-Consumer-Electronics/MarketWatch/Pages/Internet-Enabled-Consumer-Electronics-Devices-to-Enjoy-50-Percent.aspx

[10] iSuppli: Smartphones to Account for Majority of Cellphone Shipments by 2015. http://www.isuppli.com/Mobile-and-Wireless-Communications/News/Pages/Smartphones-to-Account-for-Majority-of-Cellphone-Shipments-by-2015.aspx

[11] iSuppli: Shipments of Internet-Enabled Consumer Devices to Exceed PCs in 2013. http://www.isuppli.com/Home-and-Consumer-Electronics/News/Pages/Shipments-of-Internet-Enabled-Consumer-Devices-to-Exceed-PCs-in-2013.aspx