VIRTUAL CELL AND CO-CHANNEL INTERFERENCE

There are a few examples of true co-channel interference that come to mind that I have seen while working with single channel architecture and Virtual Cell technology. I know that a major hang-up amongst wifi engineers is that it is hard to understand how single channel architecture isn't causing co-channel interference in its native form. This essay is being written not to explain how single channel architecture and Virtual Cell work without causing CCI, but to show examples of how misconfigurations and/or malfunctions can cause true CCI in a SCA environment.

We were recently called out to a school that had reported trouble with devices losing connectivity and overall poor performance in the library. Before leaving for the site we checked the configuration of the extended service set profiles and made sure the Virtual Cells were consistent and in working order. We also checked radio configurations on the two access points that serviced the library. Everything on the controller and APs appeared to be in order.

Upon arrival at the school we discussed the issues with the librarian and understood that she was truly experiencing some sort of service interruption and not a typical user error or training issue. I opened my laptop which immediately connected to the wireless network without issue. Again, everything appeared to be working fine. I opened up MetaGeek inSSIDer and quickly noticed something out of order. I was expecting to see two virtual basic service set identifiers per band, one for each SSID that we broadcast. Keep in mind that even though there are two APs in the area, there only appears to be one AP with two BSSIDs per band due to the use of Virtual Cell technology. What I found interesting was there was a third BSSID being broadcast from the 80 MHz VHT radio on U-NII-3 channel 155. Not only was there a third BSSID, but the BSSID was identifying itself as 00:00:00:00:00:00. This BSSID was also using the exact same SSID, LCPS-OPEN, as our open network. Because there was another SSID with the same name but different BSSID, co-channel interference was occurring. I went back and double checked the configuration on the controller, and again, everything checked out. At this particular time there were few connections in the area but I did notice an unusually high channel utilization that wasn't an accurate representation of the client load. I was slightly confused as to what was actually going on since nothing in the configuration indicated an inconsistency within the Virtual Cell. An inconsistency in the Virtual Cell would have been easy to spot within the extended service set profile. The inconsistency would have shown itself by the presence of multiple BSSIDs where one BSSID would be assigned to all radios per band in that specific extended service set (channel striping nor channel layering were being used in this area).

I fired up Wireshark to get a better idea of what was going on and to validate what I was seeing within inSSIDer. After only a few seconds of monitoring I was able to see that there was a wlan.bssid identifying itself as 00:00:00:00:00:00. I stopped the capture and took a deeper look into a management frame and verified that the transmitter and source address were actually 00:00:00:00:00:00 and also found that the SSID parameter was set to "SSID: LCPS-OPEN". I now had proof that an issue was occurring on one of the two access points in the library and co-channel interference was the problem.

I had a fairly good feeling that I could remedy the problem by rebooting the misbehaving access point. I started another packet capture and performed a reboot of the AP all while monitoring the packet capture. I watched the faulty wlan.bssid disappear as the radios cycled. I then watched as the

access point provisioned its radios. When I verified that the radios were again in a functional state, I checked my packet capture and found that the faulty BSSID was no longer there.

I asked the librarian for a few of the mobile devices to test out the wireless now that the issue appeared to be fixed. A few TCP throughput tests and a channel utilization check by the Aircheck G2 indicated that everything was back in order.