Appendix B - Roaming

Client Roaming Decision Making

In a typical wifi network it's the clients that decide whether to roam or not based upon how much the current connection to the AP has degraded. Obviously roaming has an impact on client traffic because of the time taken for a client to scan other wifi channels for alternative APs, then re-associate, and then authenticate to the new AP. There is often also a period before starting to roam when the connection will be a bit iffy whilst the client is thinking about whether to hop or not.

Although different clients have differing factors to consider before roaming there are some common situations that typically cause a roam to occur:

- *Maximum data retry count is exceeded*—Excessive numbers of data retries are a common roam trigger.
- *Low received signal strength indicator (RSSI)*—A client device can decide to roam when the receive signal strength drops below a threshold. This roam trigger does not require active client traffic in order to induce a roam.
- *Low signal to noise ratio (SNR)*—A client device can decide to roam when the difference between the receive signal strength and the noise floor drops below a threshold. This roam trigger does not require active client traffic in order to induce a roam.

After adding all these factors up the client will then make up it's mind whether the time has arrived for it to start looking round for a better AP to hop across to.

Layer 2, Layer 3, and how network data gets from A to B

At this stage it's worth having a quick refresher on the various stages of sending data in a network. That's best explained if we first understand the concepts of Layer 2 network traffic and Layer 3 traffic. In case the terms Layer 2 and Layer 3 are new to you, to remind you the OSI multilayer model is a conceptual model that characterizes and standardizes the various functions of a communication system into a set of 7 abstraction layers. At the bottom (layer 1) is the physical hardware for the network, e.g. the network cards and cables or 2.4GHz shortwave radio. At the top of the layers is layer 7 which is the programme running on the device that actually uses this data.

Layer 2 in this model refers to the Data Link Layer and sits above the Physical Layer (the actual networking hardware). The data link layer provides node-to-node data transfer by detecting and possibly correcting errors that may occur in the physical layer. So, in our conversation here, it pertains to the links the data needs to travel through between origination and the end receiver on the SAME network.

Network layer 3 translates logical network addresses into physical machine addresses. In a common IP network every node has an address which permits nodes to transfer messages to other nodes by providing the content of a message and the address of the destination node and then letting the network find the way to deliver ("route") the message to the destination node. So layer 3 covers the communication from end to end even if each end is on a DIFFERENT network (for example when your PC talks to an internet web site).

So layer 2 only knows about the machine addresses of devices on the same network. Layer 3 is the IP layer and understands the concept of data travelling between different

networks.

Perhaps it would help you to understand the difference between layer 2 and layer 3 if we have a typical real world example? Imagine an ftp server running on a PC connected by a lan cable to your office switch. The server wants to send a stream of data to a wireless client (let's say a laptop) which is wirelessly connected to an access point also connected to this same switch.

The transmission of data from server to client would go something like this:

Server wants to send a packet to the client. The server knows the IP address of the client so it places that in the layer 3 packet. The server now needs to place the MAC Address (the low level address of the client) in the layer 2 packet however, initially, it doesn't know what that is. So it first of all sends out something called an ARP request (Address Resolution Protocol) to the network with the destination IP. Assuming both parties are on the same network the other the machine with that IP will reply back to the sender with its MAC address. In the process of this packet toing and froing the switch in the middle learns which ports to send data to in order to reach each party in the conversation.

Now the server sends the data packet to the switch. The switch looks up the destination MAC address in its routing table and then send the packet onto the correct LAN port. From there to the access point which also checks it's own cached routing table and hence onto the wifi client.

So the switch is a key component in ensuring that data reaches the destination client. If the routing table in the switch is incorrect or out of date then the data wont get from A to B; it gets lost at the switch. If the data fails to get to it's destination, perhaps because the client has moved, then there are a number of fail safes and error checks to try and rectify the situation. Essentially this means that either the applications or intervening hardware has failed to get an acknowledgement from the destination to say the data has been received. When this happens then the sender could redo the ARP message in order to relearn the MAC address route or the switch might do a broadcast on all of it's ports to see which port to start using. Which ever method is used, the time taken to recover from a client moving locale on a network can be anything from a fraction of a second to several tens of seconds; it just depends upon the type of data, the protocol used to send the data, or the applications at each end of the communication link which are transmitting and receiving.

For example TCP traffic is a protocol where every packet from a sender has to be acknowledged by the receiver. If it's not (for example because the receiver has moved) then the sender instigates a set procedure of events including retries and, when they fail, messages to check if the receiver's moved or not. There will still be some delays whilst all this takes place but possibly only a few seconds. TCP traffic is the type used for things like web, email, file/data transfer etc... in fact TCP is the most common type if network traffic. Generally a user won't notice any hiccups if they're doing things like web surfing or checking emails. Even music streaming should be fine (since the receiver application will use a buffer for just this type of eventuality). However there's another type of data protocol called UDP. With UDP there are no expected handshakes or acknowledgements from the receiver. UDP is used for voip/sip applications. A voip call starts with the two parties negotiating the call, this is done with TCP. After this the actual voice traffic (called RTP) is sent using UDP where no acknowledgements are used to ensure receipt. The reason for this is down to traffic congestion and speed. In a large voip application where a voip exchange might, at any one time, be fielding thousands of calls then there just isn't the time or bandwidth to cope with acknowledgement messages. So it's obvious, if the

application is a voip call in progress, then there might well be a pause or disruption in the RTP (voice) stream when the client relocates on the network.

So, in summary, the implications of this are very important when one considers the processes of a wifi client roaming from one access point to another. Now you understand the various stages involved in routing data on a network it's easy to see that there are potentially big problems when a wifi client moves around a wireless network. In simple terms, if a client moves from one access point to another, whilst it's sending or receiving data, then it's obvious that there will almost certainly be some hiccup in that data flow as the routing in the network and switches sorts it's self out. How long this hiccup is and what effect it has on the applications using the data will depend upon what type of data this is. General web browsing and you probably won't notice. Doing a voip call and maybe you will.

So now that's out of the way let's investigate the methods used on wifi networks to manage roaming of a client from one access point to another. So I'll explain the protocols and methods used to streamline the migration of a client on a wireless network. Then cover what can be done about the problem of the ARP and routing entries in the various switches and computers on the network being incorrect.

Methods for WiFi Roaming

Same SSID/Security and trust to luck

Okay a very common method is just to set all your access points on the same settings and hope the client will hop from one AP to the next. This method is unreliable, slow, and data almost certainly lost.

SCA/Proprietary

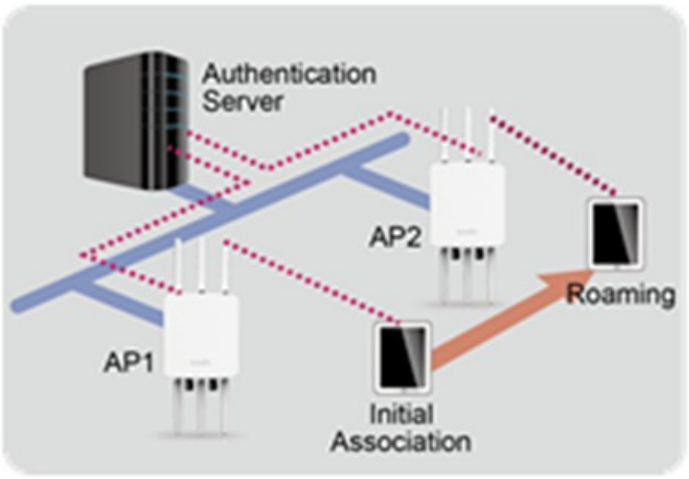
The wifi change over between access points can be fast. As mentioned above, data probably lost. Also, because all the access points run on the same wifi channel then there can be a lot of WiFi Interference so it's bad where you want wifi saturation coverage. In fact, Ubiquiti use SCA and they warn on there web site that their Unifi SCA system is not advisable where you want complete wifi coverage.

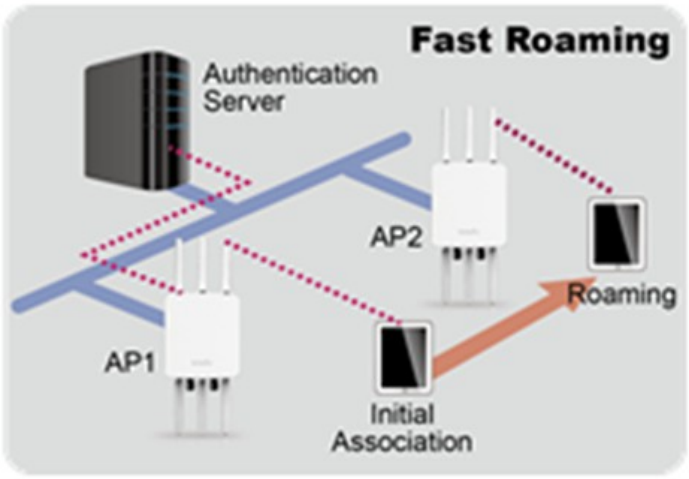
802.11r (EnGenius call Fast Roaming)

802.11r, which is the IEEE standard for fast roaming, introduces a new concept of roaming where the initial handshake with the new AP is done even before the client roams to the target AP, which is called Fast Transition (FT). The initial handshake allows the client and APs to do the Pairwise Transient Key (PTK) calculation in advance. These PTK keys are applied to the client and AP after the client does the re-association request or response exchange with the new target AP. The FT key hierarchy is designed to allow clients to make fast BSS transitions between APs without requiring re-authentication at every AP. 802.11r eliminates much of the handshaking overhead while roaming, thus reducing the hand off times between APs while providing security and QoS. This is useful for client devices that have delay-sensitive applications such as voice and video and is the key requirement for voice over Wi-Fi.

So this is this an improvement of basic SCA because each AP can be on different wifi channels. So it's good for high wifi density sites. However, on it's own, this doesn't fix the issue of loosing data as the client moves. This is just a protocol to accelerate the security processes involved when a

client connects to an access point.





Fast Handover

This isn't an official roaming protocol but it's one of the tools the EnGenius access points use to encourage client mobility. As discussed above, one of the key stages with wireless client roaming is the client making a sensible decision about when to roam. You will typically find that the quality and reliability of a wifi client connection will degrade to quite a poor state before it will trigger a decision to start investigating the option of roaming. So there will invariably be a period of data breakups prior to the client actually making the choice to move to a better access point. It would be good if the access points could monitor the signal they're seeing from the client and if it drops below a certain, settable, value, then the AP breaks the client wifi link. This then forces the client to look around and re-asses the various access point signals. In many ways this is similar to the process that a wifi client goes through to decide to roam The difference is you have control over the settings so you can set the correct signal level based upon empirical analysis. It also means you set different thresholds for each access point so you can cater for different locations and areas of differing wifi signal.

802.11k

The 802.11k standard allows clients to request neighbour reports containing information about known neighbour access points that are candidates for a service set transition. The use of the 802.11k neighbour list can limit the need for active and passive scanning. The assisted roaming feature is based on an intelligent and client optimized neighbour list. The 802.11k neighbour list is generated dynamically on-demand and is not maintained on the switch. The 802.11k neighbour list is based on the location of the clients. Two clients on the same switch but different APs can have different neighbour lists delivered depending on their individual relationship with the surrounding APs. By default, the neighbour list contains only neighbours in the same band with which the client is associated. However, the dual-list configuration allows 802.11k to return neighbours in both bands. Clients send requests for neighbour lists only after associating with the APs that advertise the RRM capability information element (IE) in the beacon. The neighbour list includes information about BSSID, channel, and operation details of the neighbouring radios.

tbh this is not actually a roaming protocol but, when used with 11r, it adds extra reliability since the client doesn't waste time looking for APs. It also reduces wifi congestion because the client it's having to send out redundant exploratory broadcasts to search for the next access point. One problem though is not all wifi clients support 802.11k/r. It IS becoming more popular so some of the

newer tablet and phone products are starting to support these protocols. However even in up to date products support is not universal and almost certainly absent from older equipment.

How Does Neutron help Roaming?

WiFi Features

To reduce the overhead intrinsic in WiFi security and QoS during the handoff process, the Neutron package supports 802.11r, also called “Fast Transition” (FT), to allow a roaming client to initialize a handshake with a new AP before it roams to the target AP. The core idea behind this is to use FT key hierarchy to allow clients to make fast BSS transitions between APs within the same ESS and mobility domain without re-authentication required at every AP.

The Neutron system now also supports 802.11k which is designed to allow clients to quickly identify nearby APs that are available for roaming. So when a client senses the signal strength getting weaker from current AP and needs to prepare hand-off to another AP, this mechanism allows the client to know the best candidate AP to roam to.

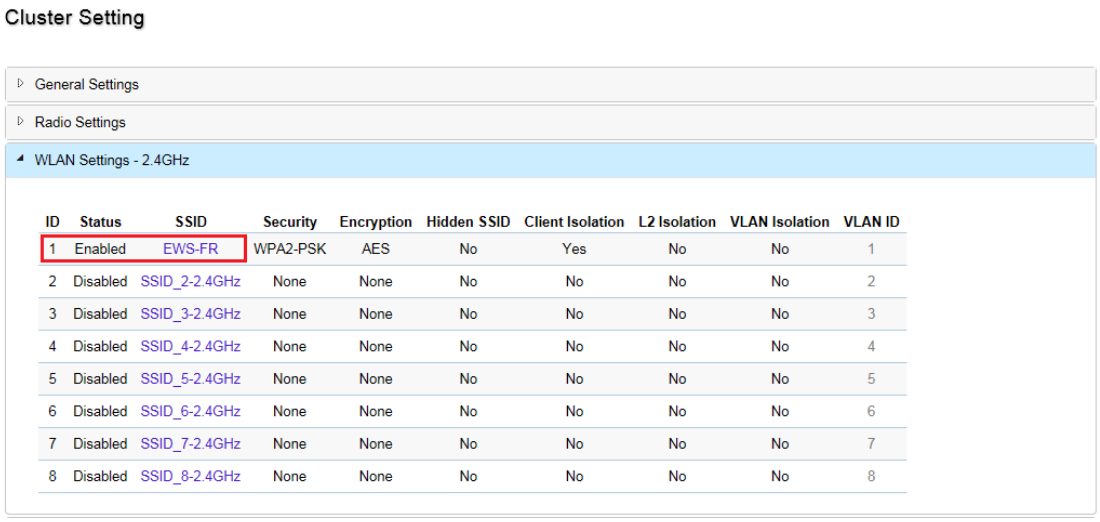
The following explains further the WLAN operations performed by the EnGenius Neutron system APs to accommodate this fast roaming feature:

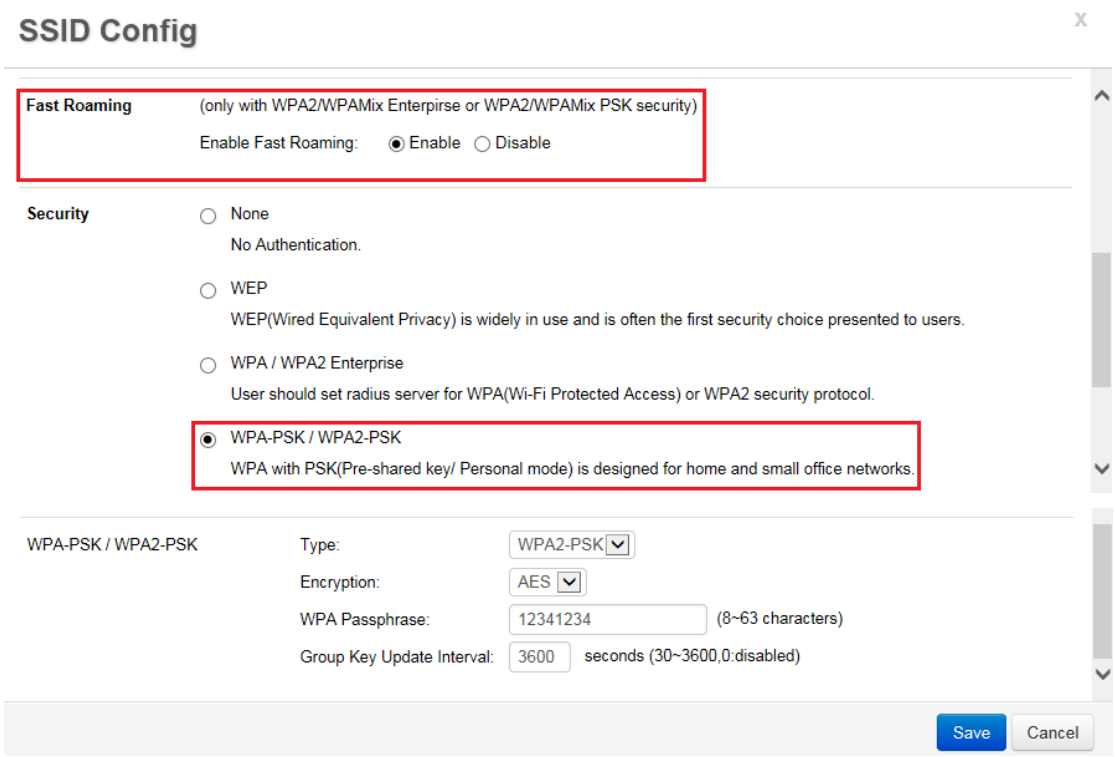
Fast Roaming is Enabled

The fast roaming feature is supported for the first SSID profile per radio with security types WPA2/WPA-Mixed PSK and WPA2/WPA-Mixed Enterprise. The following table lays out the requirements for applicable security types.

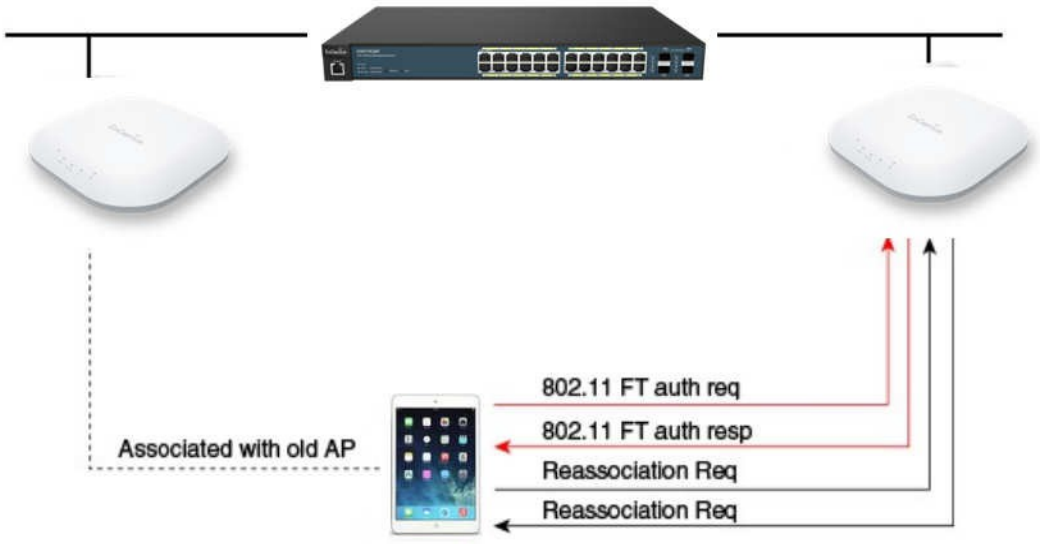
PMKSA Caching	**802.11k/r**	**Auth Server**
WPA2-Enterprise	WPA2-Enterprise	RADIUS
WPA-Mixed Enterprise	WPA-Mixed Enterprise	
	WPA2/ WPA-Mixed -PSK	RADIUS not needed

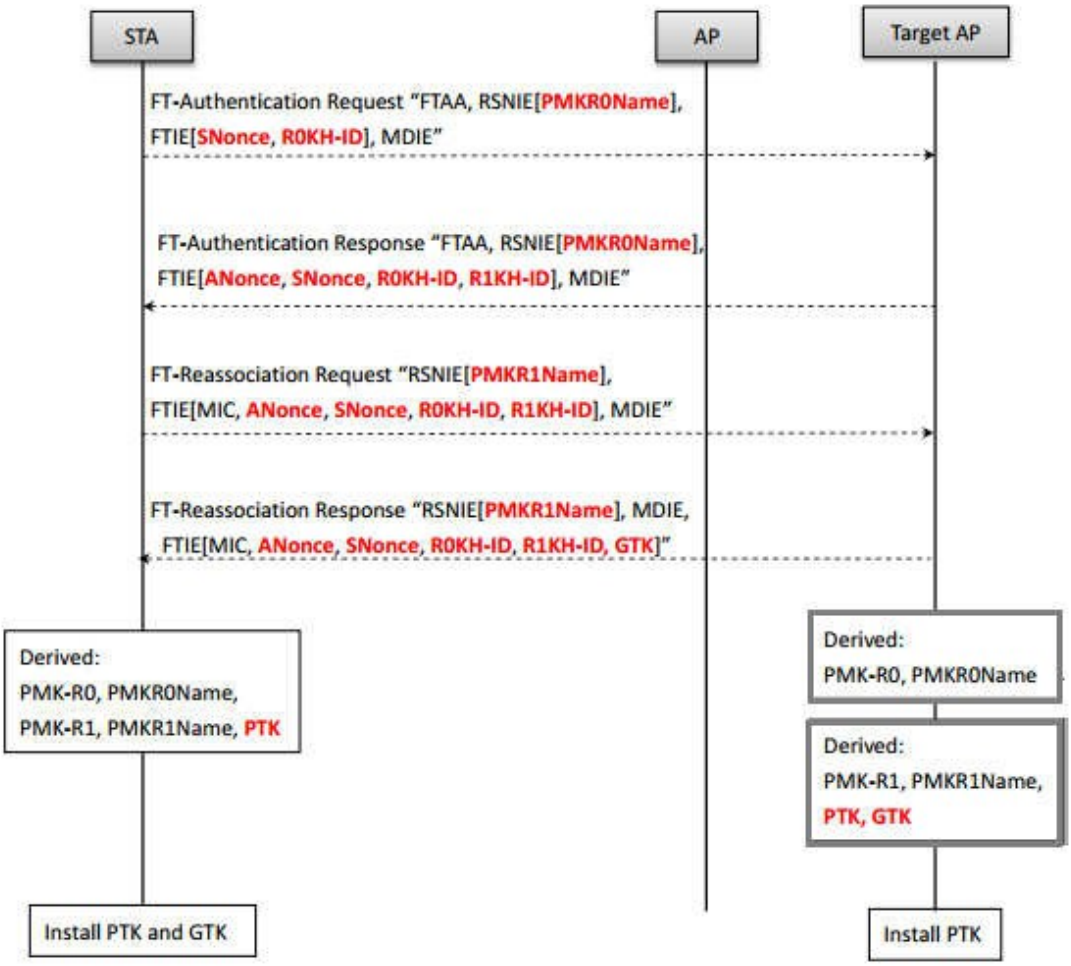
The following show a cluster setting on the Neutron controller as an example for Fast Roaming configuration:





802.11k/r-compliant Client Roaming





Before actual handoff takes place, the client has been advised of the best candidate AP for roaming. Once the new AP is identified, the various FT operations then allow client the to perform handshake with the selected new AP. The whole process is then completed before the clients roams to another AP so the process of re-authentication isn't required.
The follow table lists some well-known 802.11k/r client types as of now:

iOS device	802.11k/r support	iOS 6 and later supported methods	Pre-iOS 6 supported methods
iPad Air 2	Yes	FT, PMKID caching	Not applicable
iPad mini 3			
iPhone 6			
iPhone 6 Plus			
iPhone 5s			
iPhone 5c			
iPad Air			
iPad mini with Retina display			
iPad (4th generation)			
iPad mini			
iPhone 5			
iPod touch (5th generation)			
iPad (3rd generation)	Yes	FT, PMKID caching	PMKID caching
iPhone 4s			
iPad (2nd generation) and earlier	No	PMKID caching	PMKID caching
iPhone 4 and earlier			
iPod touch (4th generation) and earlier			

Non-802.11k/r Client Roaming

If the site uses clients which do not support 802.11k/r then obviously the Neutron controller is unable to use these protocols for the migration process. However is WPA2-Enterprise is selected for wireless security then the EnGenius APs can use 802.11i-based PMKSA caching method to improve the speed of roaming. Under this scenario, non-802.11k/r compliant client can hand off to adjacent new AP within the same ESS without re-authentication.

Upon client completing authentication with the RADIUS server through the current AP, PMKSA is created and, if fast roaming is enabled, this information is distributed through the network to adjacent supporting APs. This means, when a client prepares to roam, its surrounding APs have already been advised of the corresponding PMKSA security information

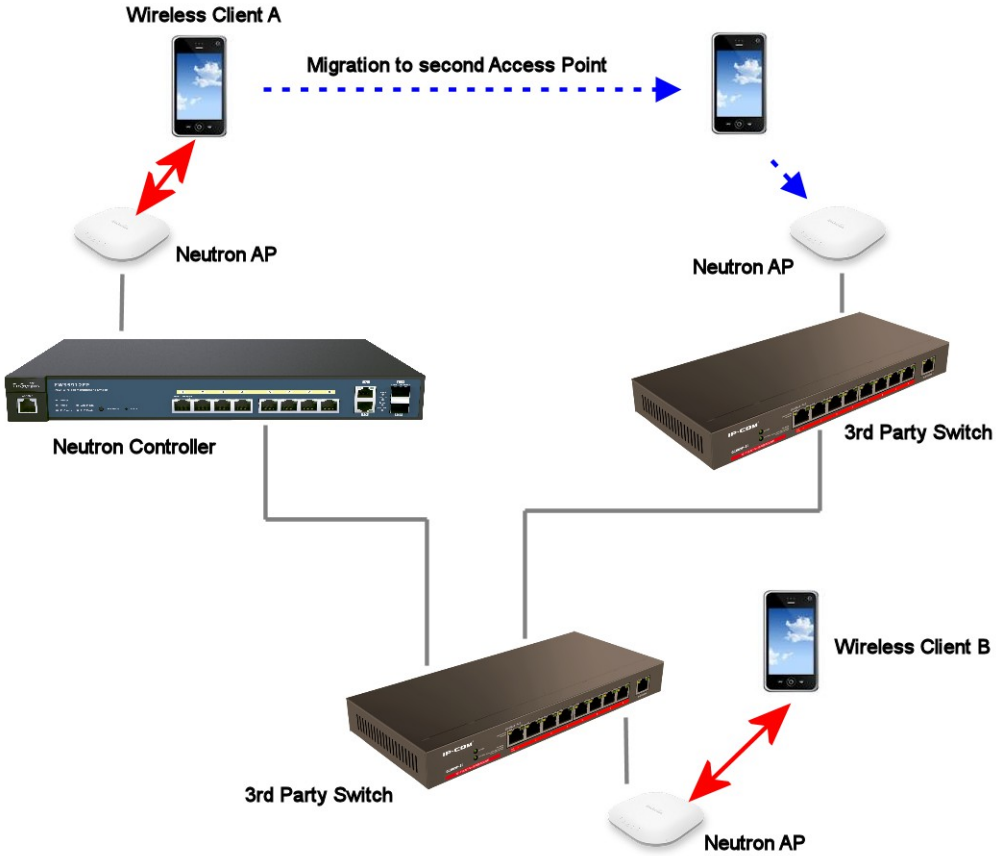
Fast Roaming is Disabled (Default)

Under this circumstances client roaming will migrate but without the advantages of PMKSA caching, FT mechanisms, or 802.11k info sharing for best candidate AP selection. It will solely depend upon the client's decision about when to roam and to which AP to connect to. If a RADIUS server is used for wireless security then authentication will be required again upon roaming.

Switch Routing

As explained above, there are two sides to the roaming issue: There's the process of the wifi 'roam' from AP1 to AP2, things like the EnGenius Fast Roaming (minimum RSSI) and 802.11k/r are the tools to make that happen correctly. Then there's the ARP and MAC address stuff. If the access points involved are connected to the Neutron management switch then the routes in the switch are automatically updated as part of the migration process – no problems then. However what if there are 3rd party switches in the chain. For example what if AP1 is connected to a Neutron switch on your network but AP2 is connected to a separate 3rd party PoE switch connected elsewhere to the network? In this case, by default, the network as a whole might not be aware of the client migration.

Consider the application below with two handheld clients connected to your sites network consisting of several Neutron access points connected to various PoE switches, one of which is the Neutron management switch:



So the issue is, when client A moves between access points then the ARP/routing tables on the 3rd party switches might not be fully updated. This means that voice traffic going from client B to A might be interrupted. Obviously, given time, one of the two clients will realise something's gone wrong and take steps to resolve the issue. With simple web traffic it's not a problem. With voip traffic it might. One fix is to use IGMP Snooping. If the 3rd party switches support IGMP Snooping (for which you would invariably need a switch with Layer 2 management tools) then, with this enabled, the switch can eavesdrop on the multicast traffic being generated in the AP-EWS switch communications and hence relearn where the client's moved to. One thing to ensure though is that the 3rd party switches are set to NOT drop unknown multicast address function and also you must enable flooding for unknown multicast traffic.