Few Words - IP Prefix Lists

Ip prefix-list provides the most powerful prefix based filtering mechanism Here is a quick little tutorial on Prefix-lists for you. A normal access-list CANNOT check the subnet mask of a network. It can only check bits to make sure they match, nothing more. A prefix-list has an advantage over an access-list in that it CAN check BOTH bits and subnet mask - both would have to match for the network to be either permitted or denied. For checking bits a prefix list ALWAYS goes from left to right and CANNOT skip any bits. A basic example would be this: 172.16.8.0/24 If there is only a / after the network (no le or ge) then the number after the / is BOTH bits checked and subnet mask. So in this case it will check the 24 bits from left to right (won't care about the last 8 bits) AND it will make sure that it has a 24 bit mask. BOTH the 24 bits checked and the 24 bit subnet mask must match for the network to be permitted or denied. No we can do a range of subnet masks also that could be permitted or denyed: 172.16.8.0/24 ge 25 If we use either the le or ge (or both le and ge) after the /, then the number directly after the / becomes ONLY bits checked and the number after the ge or le (or both) is the subnet mask. So in this case we are still going to check the first 24 bits of the network from left to right. If those match we are then going to check the subnet mask, which in this case can be GREATER THAN OR EQUAL TO 25 bits - meaning that as long as the first 24 bits of the network match the subnet mask could be 25,26,27,28,29,30,31,or 32 bits. They would all match. We can also do: 172.16.8.0/24 le 28 Again this will check the first 24 bits of the network to make sure that they match. Then it will check to make sure that the subnet mask is LESS THAN OR EQUAL TO 28 bits. Now this isn't going to be 28 bits down to 0 bits, the subnet mask can't be any lower than the bits we are checking. So the valid range of subnet masks for this one would be 28 bits down to 24 bits (24,25,26,27,and 28). All of those would match. We can also do both ge and le: 172.16.8.0/24 ge 25 le 27 Here again we are checking the first 24 bits to make sure they match. Then our subnet mask must be GREATER THAN OR EQUAL TO 25 bits LESS THAN OR EQUAL TO 27 bits. Meaning that 25,26,and 27 bit subnet masks would match. Now for a couple of examples: If we have the following networks: 172.16.8.0/28 172.16.8.16/28 172.16.8.32/28 172.16.8.48/28 172.16.8.64/28 We could permit all of these networks with on prefix-list statement: 172.16.8.0/24 ge 28 le 28 This will check the first 24 bits to make sure they match. All of these networks have 172.16.8 as the first 24 bits, and it won't care what is in the last 8 bits. Then it will check to make sure that the subnet mask is GREATER THAN OR EQUAL TO 28 bits LESS THAN OR EQUAL TO 28 bits - the only number that works for this is 28 bits. So the first 24 bits in the network must match and it has to have a 28 bit subnet mask. All 5 of our networks would match for this. We could be even more precise with this and use: 172.16.8.0/25 ge 28 le 28 If we take a look at our 4th octects we will see that for all of them the 128 bit is off so we can check that bit also (25 bits total we are checking). 0 -- 0 0 0 0 0 0 0 0 16 - 0 0 0 1 0 0 0 0 32 - 0 0 1 0 0 0 0 0 48 - 0 0 1 1 0 0 0 0 64 - 0 1 0 0 0 0 0 0 This would be closer to permitting the 5 networks that we have. We could also permit only the classful networks. The first thing that we need to do is figure out exactly what a classful network is. For a class A network we know that it has to have an 8 bit mask and must be between 0 and 127 in the first octect. If we break down 0 and 127 we get: 0 --- 0 0 0 0 0 0 0 0 127 - 0 1 1 1 1 1 1 1 For the first octect of a class A network the first bit has to be a 0, it must be off. So we can do a prefix-list like this: 0.0.0.0/1 ge 8 le 8 In our first octet the first bit is a 0 (which is what it would need to be to be class A), with the /1 we have we are ONLY checking the first bit to make sure it's a 0 (meaning it would be a class A network 0 - 127). We are then making sure that this class A network actually has a class A subnet mask of 8 bits, and only 8 bits would match. For the class B's we need to make sure that they have a 16 bit subnet mask and that they are in the range of 128 - 191 in the first octet. If we break down 128 and 191 we get: 128 - 1 0 0 0 0 0 0 0 191 - 1 0 1 1 1 1 1 1 The first two bits are what we are going to care about. We need to make sure that the first two bits in the first octet are 1 0 . The first number that we can use as our standard we are checking against is 128 - 128 has a 1 0 as the first two bits in its first octet. 128.0.0.0/2 ge 16 le 16 So we are checking the first two bits to make sure the network has a 1 0, meaning that it must be in the range of 128 - 191. We are then going to check to make sure that it has the classful 16 bit mask, and ONLY a 16 bit mask. Finally we have the class C networks. Class C networks are in the range of 192 - 223 and they must have a 24 bit mask. If we break down 192 and 223 we get: 192 - 1 1 0 0 0 0 0 0 223 - 1 1 0 1 1 1 1 1 The first 3 bits in the first octet are what we care about. 192 would be the first number we can put in that first octect that will have 1 1 0 as its first 3 bits. 192.0.0.0/3 ge 24 le 24 We are going to check the first 3 bits of the octet and make sure that its 1 1 0 meaning that it has to be in the range of 192 - 223 being class C, then we are going to check to make sure it has a class C classful subnet of 24 bits. Finally how to permit or deny any could be very helpful since a Prefix-list just like an Access-list has an implicit deny at the end: 0.0.0.0/0 le 32 This is 'any' for a prefix-list. It says check 0 bits; I don't care what any of the bits are. It also says that the subnet mask can be 32 bits or less (down to the number of bits we are checking) down to 0. So we aren't going to check any bits and the network can have a subnet mask of anything between 0 and 32 bits. This would be 'any'. Now for your Prefix-list: In the 3rd Octet we have 1, 4, and 5. We'll break these down to binary to see if we can summarize these into one line:

1 - 0 0 0 0 0 0 0 1 4 - 0 0 0 0 0 1 0 0 5 - 0 0 0 0 0 1 0 1 For a Prefix-list we need to go from the left to the right and we can't skip bits. So for these three networks we would need to stop at the 8 bit since it is the last bit from left to right that is the same. This would give us 3 bits that are different, or 8 possible networks. We only have 3 of the 8 possible networks and we should not permit or deny more than we actually have. We should be as specific as possible. If we leave the 91.86.1.0/24 alone by itself it will give us a Prefix-list of: 91.86.1.0/24 This will check the first 24 bits from left to right to make sure that they match, and it will also check to make sure that it has a 24-bit subnet mask. For the 4 and 5 networks we can permit or deny both of those with one line. If we take a look at 4 and 5 again we can see that all of the bit's match down to the 2 bit. This would leave 1 bit that doesn't match, which would give us 2 possible networks, both of which we have. The Prefix-list to permit or deny both 4 and 5 would be: 91.86.4.0/23 ge 24 le 24 This will check the first 23 bits from left to right. The 24th bit could either be off, which would give us 4, or it could be on which would give us 5. Since we have the ge and le involved the /23 is only bits checked. The ge and le specify that our subnet mask must be greater than or equal to 24-bits and less than or equal to 24-bits which means that the subnet mask must be 24-bits for both possible networks.