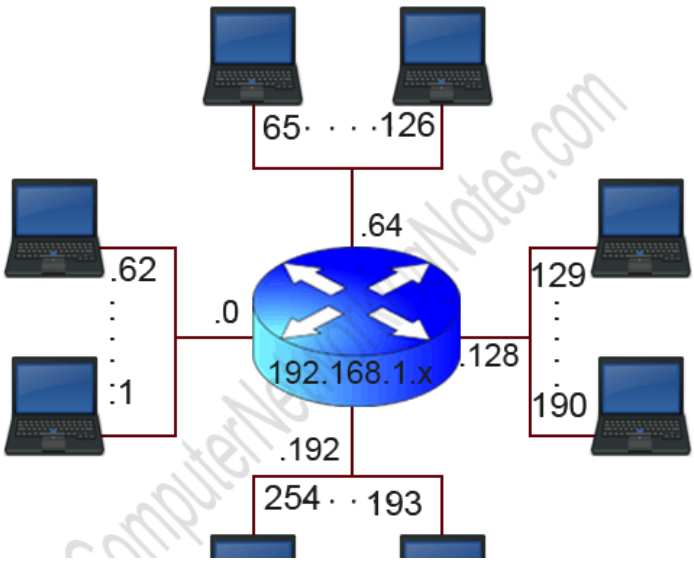
កុំព្យូទ័រណែនាំទើក

និពន្ធដោយ
បណ្ឌិត អ៊ុក ឃាន
២០២១

គាំទ្រថវិកាលើការរៀបរៀង និពន្ធ និងកែលម្អដោយ
មូលនិធិស្រាវជ្រាវ កំនិតច្នៃប្រឌិត និងនវានុវត្តន៍

កុំព្យូទ័រណេតវើក



និពន្ធដោយ
បណ្ឌិត អ៊ុក ឃាន

២០២១

គាំទ្រថវិកាលើការរៀបរៀង និពន្ធ និងកែលម្អដោយ
មូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍

សាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ

មហាវិទ្យាល័យវិទ្យាសាស្ត្រ

ដេប៉ាតឺម៉ង់ព័ត៌មានវិទ្យា

គណៈកម្មការនិពន្ធ

ឯកឧត្តមបណ្ឌិត **អ៊ុក ឃាន** ទីប្រឹក្សាក្រសួងអប់រំ យុវជននិងកីឡា

សាស្ត្រាចារ្យព័ត៌មានវិទ្យានៃសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ

គណៈកម្មការត្រួតពិនិត្យ

លោក **ជី គួច** ប្រធានដេប៉ាតឺម៉ង់ព័ត៌មានវិទ្យា

លោក **ប៉ុក លក្ខ្មុនី** អនុប្រធានដេប៉ាតឺម៉ង់ព័ត៌មានវិទ្យា

លោក **ធូ រេណូ** ប្រធានការិយាល័យស្រាវជ្រាវ

បុព្វកថា

ដំណើរអភិវឌ្ឍនៃព្រះរាជាណាចក្រកម្ពុជានៅក្នុងយុគសម័យទំនើបនេះ ជាមេរៀនដ៏ជោគជ័យបំផុតមួយ ដែលចាប់ឫសគល់ចេញពីការបញ្ចប់របបប្រល័យពូជសាសន៍ ការបញ្ចប់សង្គ្រាម ការផ្សះផ្សារជាតិ ការកសាងមូលដ្ឋានរឹងមាំនៃសន្តិភាពនិងស្ថេរភាព និងការអភិវឌ្ឍសេដ្ឋកិច្ច។ នៅក្រោយពេលដែលសន្តិភាពត្រូវបានកើតឡើងដោយបរិបូរណ៍នៅឆ្នាំ១៩៩៨ កម្ពុជាទទួលកំណើនសេដ្ឋកិច្ចខ្ពស់ គឺប្រមាណ៨% ក្នុងមួយឆ្នាំ។ លើសពីនេះទៀត អត្រានៃភាពក្រីក្រត្រូវបានកាត់បន្ថយពីប្រមាណ៥៣% នៅឆ្នាំ២០០៤ មកនៅទាបជាង១០% នៅឆ្នាំ២០១៩។ ដំណើរនៃការអភិវឌ្ឍជាតិជាសកម្មភាពដែលបន្តទៅមុខជាប់ជានិច្ច ហើយគោលនយោបាយថ្មីៗដែលមានលក្ខណៈអន្តរវិស័យគ្របដណ្តប់ក៏កំពុងលេចរូបរាងឡើង ដើម្បីតម្រង់ទិសកម្ពុជាឆ្ពោះទៅកាន់ប្រទេសមានប្រាក់ចំណូលមធ្យមកម្រិតខ្ពស់នៅឆ្នាំ២០៣០ និងឈានឡើងជាប្រទេសមានប្រាក់ចំណូលខ្ពស់ នៅឆ្នាំ២០៥០។ ការប្រែប្រួលឆាប់រហ័សនៃនិម្មាបនកម្មពិភពលោកនិងតំបន់ រួមទាំងទំនាក់ទំនងភូមិសាស្ត្រនយោបាយ បានផ្តល់កាលានុវត្តភាពសម្រាប់ការអភិវឌ្ឍឧស្សាហកម្មនៅកម្ពុជា ដែលត្រូវបានរាជរដ្ឋាភិបាលចាត់ទុកជាមូលដ្ឋានគ្រឹះនៃកំណើនសេដ្ឋកិច្ចកម្ពុជា។ រាជរដ្ឋាភិបាលកម្ពុជាបាន និងកំពុងបន្តពង្រឹងនិងអភិវឌ្ឍវិស័យអប់រំឆ្ពោះទៅរកការស្រាវជ្រាវនិងនវានុវត្តន៍ ដើម្បីពង្រឹងសមត្ថភាពនិងជំនាញរបស់ធនធានមនុស្សនៅកម្ពុជា ឱ្យស្របទៅនឹងបរិបទថ្មីនៃការអភិវឌ្ឍ ជាពិសេសការពង្រឹងសហគ្រិនភាពក្នុងការរៀបចំម៉ូដែលធុរកិច្ចថ្មីៗ។ ដើម្បីចាប់យកកាលានុវត្តភាពពីបដិវត្តន៍ឧស្សាហកម្មទី៤ និងសេដ្ឋកិច្ចឌីជីថលដែលកំពុងផុសផុលឡើង ប្រព័ន្ធអេកូឡូហ្ស៊ីដែលបង្កលក្ខណៈអំណោយផលដល់ការបង្កើតថ្មី នវានុវត្តន៍ ការស្រាវជ្រាវ និងអភិវឌ្ឍន៍ត្រូវតែមានការកែលម្អ។

បណ្តាប្រទេសនៅទ្វីបអាស៊ីកំពុងនាំមុខក្នុងការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍ ដោយមានភាគហ៊ុនប្រមាណ៤៤% នៃការវិនិយោគទាំងមូលរបស់ពិភពលោក។ ប្រទេសចិនកំពុងបន្តកសាងហេដ្ឋារចនាសម្ព័ន្ធនៃការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍ ក៏ដូចជាសមត្ថភាពមនុស្ស។ ផ្ទុយទៅវិញ ប្រទេសនៅទ្វីបអាមេរិកខាងត្បូងនិងអាហ្វ្រិក កំពុងស្ថិតនៅឆ្ងាយពីការវិនិយោគនេះ ហើយជាលទ្ធផលប្រទេសទាំងនោះក៏ពុំមានកំណើនសេដ្ឋកិច្ចគួរឱ្យកត់សម្គាល់ដែរ។ ទុនវិនិយោគសរុបលើការស្រាវជ្រាវនិងអភិវឌ្ឍរបស់ប្រទេសនៅទ្វីបអាមេរិកខាងត្បូងនិងអាហ្វ្រិក មានប្រមាណ៥%នៃការវិនិយោគទាំងមូលរបស់ពិភពលោក ក្នុងពេលដែលតំបន់ទាំង២នេះមានប្រជាជនប្រមាណ២០%នៃប្រជាជនពិភពលោក។ ប្រទេសចំនួន៦ដែលមានលំដាប់ខ្ពស់ជាងគេនៅក្នុងការវិនិយោគលើការស្រាវជ្រាវនិងអភិវឌ្ឍ រួមមានសហរដ្ឋអាមេរិក ចិន ជប៉ុន អាល្លឺម៉ង់ ឥណ្ឌា និងកូរ៉េខាងត្បូង ដែលស្មើនឹងប្រមាណ៧០%នៃទុនវិនិយោគសរុបរបស់ពិភពលោក។

តើចំណេះដឹង ផលិតផល និងសេវាកម្មថ្មីទាំងនេះកើតឡើងពីអ្វី? ហើយកើតឡើងដោយរបៀបណា? ព្រះរាជាណាចក្រកម្ពុជាកំពុងតែកសាងមូលដ្ឋានសម្រាប់ការត្រៀមខ្លួនទទួល និងប្រកួតប្រជែងក្នុងយុគសម័យបដិវត្តឧស្សាហកម្មទី៤ នៅក្នុងសេដ្ឋកិច្ចដែលផ្អែកលើពុទ្ធិ ហើយដែលប្រការនេះ

ចាំបាច់តម្រូវឱ្យពលរដ្ឋកម្ពុជា ត្រូវក្លាយខ្លួនជាពលរដ្ឋឌីជីថល ពលរដ្ឋសកល និងពលរដ្ឋដែលប្រកបដោយការទទួលខុសត្រូវ ដែលមានសមត្ថភាពក្នុងការផលិត ចែកចាយ និងប្រើប្រាស់ពុទ្ធិដើម្បីទទួលមនុញ្ញផល និងរួមចំណែកក្នុងកំណើន។ ធនាគារពិភពលោកបានធ្វើការកត់សម្គាល់តាំងពីឆ្នាំ ២០០២នូវបម្លាស់ប្តូរនៃមូលដ្ឋានសេដ្ឋកិច្ច ពីសេដ្ឋកិច្ចដែលពឹងផ្អែកលើកម្លាំងពលកម្ម និងធនធានអតិកម្ម (Labour and Resource Based Economy) ទៅកាន់សេដ្ឋកិច្ចដែលពឹងផ្អែកលើពុទ្ធិ (Knowledge Based-Economy) ដែលក្នុងន័យនេះ ពុទ្ធិគឺជាគន្លឹះនៃការអភិវឌ្ឍ។ អាស្រ័យហេតុនេះ នៅលើគន្លងដែលកម្ពុជាកំពុងធ្វើដំណើរឆ្ពោះទៅកាន់សេដ្ឋកិច្ចឌីជីថល សង្គមកម្ពុជាត្រូវតែមានសមត្ថភាពក្នុងការផលិត ជ្រើសរើស បន្សាំ បង្កើតមុខរបរ និងប្រើប្រាស់ពុទ្ធិ ដើម្បីរក្សានិរន្តរភាពនៃកំណើន និងកែលម្អជីវភាពរស់នៅ។ សមត្ថភាពទាំងនេះ អាចកើតឡើងនៅពេលពលរដ្ឋកម្ពុជាមានឱកាសក្នុងការទទួលបានបទពិសោធន៍ពីការស្រាវជ្រាវ ការបណ្តុះគំនិតច្នៃប្រឌិត និងការស្វែងរកនវានុវត្តន៍។

កំណែទម្រង់វិស័យអប់រំ គឺជាការត្រួសត្រាយមាគ៌ាសម្រាប់ដំណើរឆ្ពោះទៅកាន់សង្គមប្រកបដោយពុទ្ធិ និងប្រជាពលរដ្ឋប្រកបដោយភាពរស់រវើក។ តាមរយៈមូលដ្ឋានអប់រំ សង្គមប្រកបដោយពុទ្ធិនឹងប្រមូលផ្តុំ បង្កើត និងចែករំលែក ទៅកាន់សមាជិកក្នុងសង្គមនូវសម្បទាអប់រំ ពិសេសគឺពុទ្ធិសម្បទាក្នុងបុព្វហេតុនៃមនុស្សជាតិនិងឧត្តមប្រយោជន៍នៃប្រទេស។ សង្គមប្រកបដោយពុទ្ធិ គឺពុំគ្រាន់តែជាសង្គមដែលសម្បូរព័ត៌មានប៉ុណ្ណោះទេ តែជាសង្គមដែលប្រជាពលរដ្ឋអាចធ្វើបរិវត្តកម្មពីព័ត៌មានទៅជាមូលធនប្រកបដោយប្រសិទ្ធភាព។ ការរីកចម្រើនទៅមុខដ៏លំដាប់នៃបច្ចេកវិទ្យានិងតំណភ្ជាប់ បានពង្រីកព្រំដែននៃការចូលទៅកាន់ និងការទទួលបានព័ត៌មានជាសកល ហើយដែលក្នុងន័យនេះ ការអប់រំនឹងបន្តវិវត្តទៅមុខនិងមានការផ្លាស់ប្តូរ។ សង្គមមួយដែលមានអំណាន និងរបាប់ជាបុរេលក្ខខណ្ឌនៃជីវភាពប្រចាំថ្ងៃនៃប្រជាពលរដ្ឋ ពេលនោះបំណិននៃអំណាន និពន្ធ និងការគណនាលេខនព្វន្ត គឺជាចលករនៃការរៀនរបស់សិស្ស។ ធាតុដ៏ចម្បងមួយដែលស្ថិតនៅក្នុងការកសាងសង្គមដែលប្រកបដោយពុទ្ធិគឺសៀវភៅសិក្សា ហើយការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សាជាប្រចាំ គឺជានវានុវត្តន៍នៃវិស័យអប់រំដែលនាំទៅរកការសិក្សាពេញមួយជីវិត ការអភិវឌ្ឍសម្បទាអប់រំ និងការចែករំលែកចំណេះដឹង។ មូលដ្ឋានអប់រំ ជាពិសេសគឺគ្រឹះស្ថានឧត្តមសិក្សាត្រូវមានតួនាទីដែលប្រកបដោយការឆ្លើយតប ចំពោះតម្រូវការខាងលើនេះ។ សាស្ត្រាចារ្យ អ្នកស្រាវជ្រាវ និងបុគ្គលិកអប់រំត្រូវបន្តសិក្សាជាប់ជានិច្ច តាមរយៈការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សា ហើយដែលសៀវភៅសិក្សាទាំងនេះនឹងក្លាយជាស្ពាននៃទំនាក់ទំនងរវាងនវានុវត្តន៍នៃបច្ចេកវិទ្យា និងការរៀននិងបង្រៀននៅក្នុងថ្នាក់រៀន។

សង្គមដែលប្រកបពុទ្ធិ ក៏ជាសង្គមដែលបណ្តុះឱ្យមានរចនាសម្ព័ន្ធទន់នៃសេដ្ឋកិច្ចដែលពឹងផ្អែកលើពុទ្ធិដែរ។ ឧទាហរណ៍ជាក់ស្តែងនៃបែបផែននេះរួមមាន Silicon Valley នៃសហរដ្ឋអាមេរិក សួនឧស្សាហកម្មវិស្វកម្មអាកាសយានយន្តនិងយានយន្តនៅទីក្រុង Munich ប្រទេសអាល្លឺម៉ង់ តំបន់ជីវបច្ចេកវិទ្យានៅក្រុង Hyderabad ប្រទេសឥណ្ឌា តំបន់ផលិតគ្រឿងអេឡិចត្រូនិកនិងសារគមនាគមន៍ឌីជីថលនៅទីក្រុង Seoul ប្រទេសកូរេខាងត្បូង ក៏ដូចជាសួនឧស្សាហកម្មថាមពល និងឥន្ធនគីមីសាស្ត្រនៃប្រទេសប្រេស៊ីល ហើយក៏នៅមានទីក្រុងនៃប្រទេសជាច្រើនទៀតនៅលើពិភពលោក។ លក្ខណៈសម្បត្តិ

នៃទីក្រុងទាំងនេះគឺការប្រើប្រាស់និន្នាការនៃការអភិវឌ្ឍដែលជំរុញ និងតម្រង់ទិសដោយចំណេះដឹង ហើយដែលចំណេះដឹងទាំងនោះកើតចេញជាដំបូងពីការវិនិយោគទៅលើគ្រឹះស្ថានឧត្តមសិក្សា ស្ថាប័នស្រាវជ្រាវ មជ្ឈមណ្ឌលឧត្តមភាពនៃជំនាញជាន់ខ្ពស់ ការប្រកួតប្រជែងដោយគុណាធិបតេយ្យ និងជាពិសេសគឺការបណ្តុះវប្បធម៌អំណាននិងនិពន្ធសៀវភៅ។ ល្បឿននៃការរីកចម្រើនផ្នែកពុទ្ធិ និងបច្ចេកវិទ្យាកំពុងមានសន្ទុះលឿនជាងអ្វីដែលសិស្ស និងនិស្សិតអាចទទួលបានពីគ្រូនៅគ្រឹះស្ថានសិក្សា ដែលធ្វើឱ្យគោលដៅនៃការអប់រំនៅពេលបច្ចុប្បន្ននេះ មានការប្រឈមខ្លាំងជាងពេលណាទាំងអស់។ ឧទាហរណ៍ ក្នុងមួយឆ្នាំ មានសៀវភៅជាង២,២លានចំណងជើង ត្រូវបានសរសេរនិងបោះពុម្ព ដែលក្នុងនោះ ប្រទេសចិនមាន៤៤០ពាន់ ចំណែកឯសហរដ្ឋអាមេរិកមាន៣០៥ពាន់ និងប្រទេសរុស្ស៊ីមាន១២០ពាន់ចំណងជើង។

ខណៈពេលដែលបច្ចេកវិទ្យាកំពុងរីកចម្រើនជារៀងរាល់ថ្ងៃ មធ្យោបាយសម្រាប់អំណានក៏មានច្រើនជម្រើសសម្រាប់សិស្ស-និស្សិត និងសាធារណៈជន រួមមានការអានសៀវភៅ ការអានលើឧបករណ៍អេឡិចត្រូនិក ការអានដោយប្រើទូរសព្ទវៃឆ្លាត និងការអានលើកុំព្យូទ័រ ដែលសុទ្ធសឹងជាមធ្យោបាយសំខាន់ៗដែលនាំអ្នកអានទាំងឡាយឱ្យសម្រេចគោលបំណងអានរបស់ខ្លួន។ ម្យ៉ាងវិញទៀត អំណានដោយប្រើមធ្យោបាយបច្ចេកវិទ្យាទំនើប ចំណាយពេលតិច ងាយស្រួលអាន និងជួយដល់បរិស្ថានមួយកម្រិតទៀត។ នាពេលបច្ចុប្បន្ន សិស្ស-និស្សិត និងសាធារណៈជនកម្ពុជាដែលស្រឡាញ់អំណានកំពុងតែប្រើប្រាស់មធ្យោបាយអំណានទាំងនេះ។ បើយើងក្រឡេកមើលទៅប្រទេសជឿនលឿន ទោះបីជាបច្ចេកវិទ្យារីកចម្រើនខ្លាំងយ៉ាងណា អំណានតាមរយៈសៀវភៅនៅតែមានសន្ទុះដដែល។ ម្យ៉ាងវិញទៀត បច្ចេកវិទ្យាអានបែបទំនើបតាមរយៈឧបករណ៍ទំនើប អាស្រ័យលើលទ្ធភាពនៃធនធានអប់រំឌីជីថល និងមាតិកាឌីជីថលគ្រប់គ្រាន់ដែលបានផលិត និងបង្ហោះចែកចាយសម្រាប់អំណាន។

ក្នុងបរិបទកម្ពុជា ជាពិសេសក្នុងបរិការណ៍នៃការផ្ទុះរីករាលដាលនៃជំងឺកូវីដ-១៩ ក្រសួងអប់រំ យុវជន និងកីឡា បានជំរុញឱ្យមានបរិវត្តកម្មឌីជីថលនៅក្នុងអេកូស៊ីស្តែមនៃការអប់រំ ជាពិសេសការអប់រំតាមប្រព័ន្ធអេឡិចត្រូនិកនិងការអប់រំពីចម្ងាយ ដើម្បីលើកកម្ពស់អំណាន តាមរយៈការផលិតមាតិកាឌីជីថលដែលមានភាពចំរុះ ការកសាងសមត្ថភាពផ្នែកតំណភ្ជាប់និងវេទិកាឌីជីថល ការពង្រីកវិសាលភាពនៃមជ្ឈមណ្ឌលទិន្នន័យ និងការលើកកម្ពស់គុណភាពនៃការផលិតធនធានអប់រំឌីជីថល គួបផ្សំជាមួយការចែកសន្លឹកកិច្ចការឱ្យសិស្សយកទៅរៀននៅផ្ទះ និងការចុះទៅជួបជាមួយសិស្សជាបណ្តុំនៅតាមសហគមន៍។ ក្នុងន័យលើកកម្ពស់អំណាន និងភាពសម្បូរបែបនៃធនធានសៀវភៅសិក្សា ឱ្យកាន់តែមានប្រសិទ្ធភាពនិងភាពសក្តិសិទ្ធិ និងផ្តល់ឱកាសអំណានកាន់តែច្រើនថែមទៀតដល់សិស្សានុសិស្ស និស្សិត និងសាធារណៈជន ក្រសួងអប់រំ យុវជន និងកីឡាលើកទឹកចិត្តនូវចំណុចមួយចំនួនដូចខាងក្រោម៖

១. សាស្ត្រាចារ្យ អ្នកស្រាវជ្រាវ និងបុគ្គលិកអប់រំ សូមបន្តនិងបង្កើនការបោះពុម្ពស្នាដៃបន្ថែមទៀត ដើម្បីធ្វើឱ្យធនធានសម្រាប់អំណានកាន់តែសម្បូរបែប ជាពិសេសធនធានអំណានជាខេមរភាសា

២. គ្រឹះស្ថានឧត្តមសិក្សា សូមផ្តល់លទ្ធភាពគ្រប់បែបយ៉ាង ដើម្បីឱ្យបុគ្គលិកអប់រំគ្រប់លំដាប់ថ្នាក់ និងនិស្សិតគ្រប់កម្រិតសិក្សាអាចចូលរួមអាន និងសិក្សាស្រាវជ្រាវតាមគ្រប់លទ្ធភាព ជាមួយធនធានអំណាន ជាពិសេសការរៀបចំឱ្យមានពេលវេលាសម្រាប់សហសិក្សា និងអំណានក្នុងបណ្ណាល័យ

៣. សាស្ត្រាចារ្យតាមមុខវិជ្ជា និងអ្នកស្រាវជ្រាវតាមជំនាញឬវិស័យ ត្រូវរៀបចំដំណើរការរៀនបង្រៀន និងស្រាវជ្រាវដែលមានដាក់បញ្ចូលកិច្ចការស្វ័យសិក្សា សហសិក្សា ឬការស្រាវជ្រាវបណ្ណាល័យដែលតម្រូវឱ្យនិស្សិត ត្រូវអាននិងស្រាវជ្រាវជាមួយធនធានអំណាន

៤. គ្រឹះស្ថានឧត្តមសិក្សា និងមជ្ឈមណ្ឌលស្រាវជ្រាវ ត្រូវខិតខំឱ្យអស់លទ្ធភាពក្នុងការបង្កើតបណ្ណាល័យ មជ្ឈមណ្ឌលរក្សាឯកសារ ឬមជ្ឈមណ្ឌលអប់រំឌីជីថលជាដើម ដើម្បីឱ្យបុគ្គលិកអប់រំគ្រប់លំដាប់ថ្នាក់និងនិស្សិតគ្រប់កម្រិតសិក្សាអាចទទួលបាន និងស្វែងរកប្រភពសម្រាប់អំណានកាន់តែសម្បូរបែប និងមានភាពបត់បែន ឆ្លើយតបតាមតម្រូវការអ្នកអាន

៥. និស្សិតគ្រប់កម្រិតសិក្សាត្រូវខិតខំនិងចំណាយពេលវេលាដើម្បីអាន និងចាត់ទុកវប្បធម៌និងអកប្បកិរិយាអំណានជាផ្នែកមួយ នៃពេលវេលានិងភាពស៊ីវិល័យនៃជីវិតប្រចាំថ្ងៃ

៦. បងប្អូនជនរួមជាតិ ដែលជាមាតាបិតា ឬអ្នកអាណាព្យាបាល សូមជួយជំរុញនិងបង្កលក្ខណៈកាន់តែច្រើនថែមទៀត ជាពិសេសការលៃលកចំណាយនៅក្នុងគ្រួសារសម្រាប់ការទិញសម្ភារៈសិក្សា សៀវភៅអាន និងឧបករណ៍សម្រាប់អំណានដល់កូនៗ ដែលចាត់ទុកជាការវិនិយោគមួយដ៏សំខាន់ សម្រាប់ បង្កើនចំណេះដឹង និងអនាគតរបស់ពួកគេ។

ដោយមានការគាំទ្រពីក្រសួងសេដ្ឋកិច្ច និងហិរញ្ញវត្ថុ នៅឆ្នាំ២០២០ ក្រសួងអប់រំ យុវជន និងកីឡា បានបង្កើតមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍ ដែលហៅកាត់ថា “មូលនិធិ ស.គ.ន.” និងហៅជាភាសាអង់គ្លេសថា The Research Creativity and Innovation Fund ដែលហៅកាត់ជាភាសាអង់គ្លេសថា “RCI Fund”។ គោលដៅចម្បងនៃមូលនិធិនេះ គឺរួមចំណែកលើកកម្ពស់វប្បធម៌នៃការស្រាវជ្រាវ បំផុសគំនិតច្នៃប្រឌិត និងជំរុញការធ្វើនវានុវត្ត ដើម្បីជាប្រយោជន៍ដល់វិស័យអប់រំ យុវជន និងកីឡា ដែលឆ្លើយតបទៅនឹងទីផ្សារពលកម្ម និងសាកលភាវូបនីយកម្ម។ មូលនិធិ ស.គ.ន. បានសម្រេចកំណត់ប្រធានបទ ជាអាទិភាពសម្រាប់ការគាំទ្រដោយមូលនិធិចំនួន៣ រួមមានឌីជីថលនីយកម្មសម្រាប់បដិវត្តឧស្សាហកម្ម៤.0 (Digitalization for IR.4.0) ការស្រាវជ្រាវអនុវត្តលើវិស័យកសិកម្ម (Applied Agricultural Research) និងការស្រាវជ្រាវគរុកោសល្យសតវត្សទី២១ (21st Century Pedagogy Research)។

ដោយមានការធ្វើអាទិភាពវូបនីយកម្មទៅលើទិសដៅ នៃការប្រើប្រាស់ថវិកាមូលនិធិសម្រាប់ឆ្នាំ ២០២០ ក្រសួងសេដ្ឋកិច្ច និងហិរញ្ញវត្ថុ និងក្រសួងអប់រំ យុវជន និងកីឡា បានផ្តល់ការគាំទ្រដល់ការ**រៀបរៀង និពន្ធ និងកែលម្អ សៀវភៅសិក្សា (Text book) ដែលនឹងត្រូវប្រើប្រាស់នៅកម្រិតឧត្តមសិក្សា**។ គោលបំណងនៃការរៀបរៀង និពន្ធ និងកែលម្អ សៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា គឺដើម្បីបង្កើនបរិមាណ លើកកម្ពស់គុណភាព និងពង្រីកសមធម៌នៃធនធានសិក្សាជាខេមរភាសា ជូនដល់និស្សិត

ដែលកំពុងបន្តការសិក្សា និងត្រៀមខ្លួនធ្វើការស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា។ លើសពីនេះទៀត ការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា មានគោលដៅដូចខាងក្រោម ៖

១. ឆ្លើយតបជាបន្ទាន់ចំពោះការខ្វះខាតធនធានសិក្សា ដែលជាតម្រូវការសិក្សារបស់និស្សិតនៅកម្រិតឧត្តមសិក្សា

២. លើកកម្ពស់ទំនើបភាវូបនីយកម្ម និងឧត្តមានុវត្តន៍នៃការរៀននិងបង្រៀន និងការស្រាវជ្រាវនៅលើមុខវិជ្ជា កម្មវិធីសិក្សា ឬមុខជំនាញជាក់លាក់

៣. បង្កើនភាពស៊ីជម្រៅក្នុងការកសាងវិជ្ជាជីវៈនិងបទពិសោធន៍សម្រាប់ហានៈសាស្ត្រាចារ្យ និងអ្នកស្រាវជ្រាវ

៤. រួមចំណែកដល់ការកសាងភាពជាសហគមន៍វិជ្ជាជីវៈ ការចែករំលែកបទពិសោធន៍ និងវប្បធម៌នៃការរៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា។

ក្រសួងអប់រំ យុវជន និងកីឡា បានវាយតម្លៃខ្ពស់ចំពោះការបោះជំហានប្រកបដោយមនសិការវិជ្ជាជីវៈនៃគ្រឹះស្ថានឧត្តមសិក្សា និងបុគ្គលិកអប់រំទាំងអស់ ក្នុងការរៀបចំ រៀបរៀង និពន្ធ និងកែលម្អ សៀវភៅសិក្សា ដើម្បីបង្កើនបរិមាណ លើកកម្ពស់គុណភាព និងពង្រឹងសមធម៌នៃធនធានសិក្សាជាខេមរភាសា ជូននិស្សិតដែលកំពុងបន្តការសិក្សា និងត្រៀមខ្លួនធ្វើការស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា។ សៀវភៅសិក្សាជាផ្នែកមួយនៃការទទួលស្គាល់គុណភាពអប់រំនៃគ្រឹះស្ថានឧត្តមសិក្សា និងជាធនធានសិក្សាដែលជាមូលដ្ឋានមួយដ៏សំខាន់ ក្នុងការគាំទ្រដល់ការបង្រៀន និងរៀន ហើយត្រូវមានបរិមាណគ្រប់គ្រាន់ ឆ្លើយតបទៅនឹងកម្មវិធីអប់រំ និងតម្រូវការសិក្សាស្រាវជ្រាវ។ ជាគោលការណ៍ គ្រឹះស្ថានឧត្តមសិក្សាទាំងអស់ ត្រូវមានសៀវភៅសិក្សាដែលប្រើជាគោលសម្រាប់មុខវិជ្ជានីមួយៗ។ ចំនួនសៀវភៅសិក្សាដែលគ្រប់គ្រាន់សម្រាប់ការស្រាវជ្រាវ និងការសិក្សារបស់និស្សិត ត្រូវមានយ៉ាងតិចមួយចំណងជើងក្នុងមួយមុខវិជ្ជា ហើយត្រូវតម្កល់យ៉ាងតិច២ច្បាប់នៅក្នុងបណ្ណាល័យ ឬអាចរកបានតាមប្រព័ន្ធអេឡិចត្រូនិក។ ក្រសួងអប់រំ យុវជន និងកីឡា លើកទឹកចិត្តបន្ថែមទៀតជូនដល់គ្រឹះស្ថានឧត្តមសិក្សារដ្ឋ និងឯកជនដែលបានស្នើសុំថវិកាមូលនិធិ ស.គ.ន រួច សូមចូលរួមបន្ថែមទៀតដើម្បីបង្កើនចំនួនចំណងជើងសៀវភៅ។ ចំណែកគ្រឹះស្ថានឧត្តមសិក្សារដ្ឋ និងឯកជនដែលពុំទាន់បានដាក់ពាក្យស្នើសុំថវិកាមូលនិធិ ដើម្បីរៀបរៀង និពន្ធ និងកែលម្អ សៀវភៅសិក្សានៅកម្រិតឧត្តមសិក្សា សូមរួសរាន់ចូលរួមដើម្បីជាគុណប្រយោជន៍ដល់តម្រូវការដ៏ទទូចនិងថ្លៃថ្លានៃនិស្សិតកម្ពុជាក្នុងការសិក្សា និងស្រាវជ្រាវនៅកម្រិតឧត្តមសិក្សា។

សេចក្តីបញ្ជាក់
នៃមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍

សៀវភៅសិក្សានេះជាលទ្ធផលនៃការស្នើសុំអនុវត្តថវិកាមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍ ក្នុងគម្រោង**រៀបរៀង និពន្ធ និងកែលម្អសៀវភៅសិក្សា ដែលនឹងត្រូវប្រើប្រាស់នៅកម្រិតឧត្តមសិក្សា។** សៀវភៅសិក្សានេះ ត្រូវបានរៀបរៀង និពន្ធ ឬកែលម្អដោយមានការធានាអះអាងថាជាស្នាដៃរបស់អ្នកនិពន្ធផ្ទាល់ និងបានឆ្លងកាត់ត្រួតពិនិត្យ ផ្តល់យោបល់ និងវាយតម្លៃដោយក្រុមប្រឹក្សាអប់រំ ក្រុមប្រឹក្សាស្រាវជ្រាវ ឬក្រុមប្រឹក្សាដែលមានតម្លៃស្មើនៃគ្រឹះស្ថានឧត្តមសិក្សា និងតាមរយៈកិច្ចសន្យាដែលបានធ្វើឡើង និងដែលបានតម្កល់ទុកនៅមូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍។ ពាល់ខ្លឹមសារ ការបកស្រាយ ឬរូបភាព ដែលមាននៅក្នុងសៀវភៅនេះ គឺជាជំហរនិងទស្សនៈផ្ទាល់របស់អ្នកនិពន្ធ ហើយពុំឆ្លុះបញ្ចាំង ឬជាតំណាងដល់មូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍ នៃក្រសួងអប់រំ យុវជន និងកីឡាឡើយ។

អារម្ភកថា

សៀវភៅនេះត្រូវបានរៀបរៀងឡើងក្រោមការឧបត្ថម្ភគាំទ្ររបស់មូលនិធិស្រាវជ្រាវ គំនិតច្នៃប្រឌិត និងនវានុវត្តន៍សម្រាប់ទុកជាសៀវភៅសិក្សាគោលដើម្បីបម្រើឲ្យការរៀននិងបង្រៀនមុខវិជ្ជា **កុំព្យូទ័រណេតវើក** ក្នុងដេប៉ាតឺម៉ង់ព័ត៌មានវិទ្យានៃមហាវិទ្យាល័យវិទ្យាសាស្ត្រនៅសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ។ សៀវភៅនេះផ្តល់ជូនអ្នកសិក្សាទាំងផ្នែកទ្រឹស្តីនិងអនុវត្តន៍ដែលខ្លឹមសារគោលក្នុងសៀវភៅនេះមានតាំងពីការសញ្ញាណទូទៅនៃកុំព្យូទ័រណេតវើកហូតដល់កម្រិតវិស្វកម្មផ្នែកជំនាញណេតវើកពីដំបូងហូតដល់កម្រិតខ្ពស់ដែលអាចសមល្មមឲ្យអ្នកសិក្សាផ្នែកការគ្រប់គ្រងលើប្រព័ន្ធណេតវើកទាំងឡាយមានមូលដ្ឋានគ្រឹះរឹងមាំក្នុងការសិក្សាបន្ត ឬស្រាវជ្រាវបន្ថែម ឬក្នុងការប្រកបការងារផ្សេងៗក្នុងវិជ្ជាជីវៈរបស់ខ្លួន។ មិនតែប៉ុណ្ណោះ សៀវភៅនេះក៏អាចចាត់ទុកជាឯកសារពិគ្រោះសម្រាប់អ្នកស្នេហាផ្នែកដំឡើងនិងហេដ្ឋារចនាសម្ព័ន្ធនៃប្រព័ន្ធណេតវើកផងដែរ។

សេចក្តីថ្លែងអំណរគុណ

ខ្ញុំបាទអ៊ុក ឃាន ជាទីប្រឹក្សាក្រសួងអប់រំ យុវជននិងកីឡានិងជាសាស្ត្រាចារ្យដេប៉ាតឺម៉ង់ព័ត៌មានវិទ្យានៃមហាវិទ្យាល័យវិទ្យាសាស្ត្រនៃសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ និងជាអ្នករៀបរៀងសៀវភៅសិក្សាដែលមានចំណងជើងថា "កុំព្យូទ័រណេតវើក"។ ខ្ញុំបាទសូមសម្តែងនូវគោរវកិច្ចដ៏ខ្ពង់ខ្ពស់ និងសូមគោរពថ្លែងអំណរគុណ និងដឹងគុណយ៉ាងជ្រាលជ្រៅចំពោះ៖

- ក្រសួងសេដ្ឋកិច្ច និងហិរញ្ញវត្ថុ ដែលបានផ្តល់ថវិកាមកក្រសួងអប់រំ ដើម្បីបង្កើតជាមូលនិធិសម្រាប់គាំទ្រដល់ការសរសេរសៀវភៅសិក្សាសម្រាប់កម្រិតឧត្តមសិក្សារបស់ខ្ញុំបាទនេះ។

- ក្រសួងអប់រំ យុវជន និងកីឡា ដែលមាន ឯកឧត្តមបណ្ឌិតសភាចារ្យរដ្ឋមន្ត្រី ហង់ជួន ណារ៉ុន ជាប្រមុខ ដែលលោកបានដឹកនាំបង្កើតមូលនិធិស្រាវជ្រាវ គំនិតឆ្នៃប្រឌិត និងនវានុវត្តន៍ឡើង ដើម្បីគាំទ្រទាំងស្រុងដល់ដំណើរការសរសេរសៀវភៅនេះ ដែលជាចំណែកមួយក្នុងបេសកកម្មកំណែទម្រង់វិស័យអប់រំឧត្តមសិក្សារបស់លោក។

- ឯកឧត្តមបណ្ឌិត សាន វឌ្ឍនា អនុរដ្ឋលេខាធិការក្រសួងអប់រំ យុវជន និងកីឡា ដែលក្នុងនាមលោកជាតំណាងមូលនិធិ លោកបានយកអស់កម្លាំងការចិត្ត ជួយចាត់ចែងនិងជ្រោមជ្រែងយ៉ាងពេញទំហឹង ព្រមទាំងផ្តល់ជាអនុសាសន៍និងដំណោះស្រាយនានា ដើម្បីជំនះនូវរាល់ឧបសគ្គដែលកើតមានក្នុងដំណើរការសរសេរសៀវភៅនេះ។

- ឯកឧត្តមបណ្ឌិត ជេត ជាលី សាកលវិទ្យាធិការនៃសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ ដែលបានយកអស់កម្លាំងកាយចិត្ត ព្រមទាំងពេលវេលាដ៏មមាញឹកនិងមានតម្លៃរបស់លោកក្នុងការជួយលើកទឹកចិត្ត ជួយខ្នះខ្នែងដោះស្រាយ និងផ្តល់អនុសាសន៍ដ៏ត្រឹមត្រូវ ក៏ដូចជាបង្កលក្ខណៈងាយស្រួលនានា ទាំងផ្នែករដ្ឋបាល និងទាំងផ្នែកបច្ចេកទេស ដើម្បីឲ្យដំណើរការរៀបរៀងសៀវភៅនេះបានប្រព្រឹត្តិទៅដោយរលូនរហូតចប់ចុងចប់ដើម។

- លោក លួន ពេជ្របូរី និង អស់លោក-លោកស្រី ជាក្រុមការងារសម្របសម្រួលទាំងអស់របស់មូលនិធិស្រាវជ្រាវ គំនិតឆ្នៃប្រឌិត និងនវានុវត្តន៍ ដែលបានយកអស់កម្លាំងកាយចិត្ត ជួយឈឺឆ្អាលក្នុងការដោះស្រាយបញ្ហាប្រឈមនានាយ៉ាងសស្រាក់សស្រាំជាទីបំផុត ដោយមិនគ្នូញត្អែរ និងមិនខ្លាចនឿយហត់ ដើម្បីឲ្យការសរសេរសៀវភៅនេះប្រព្រឹត្តិទៅដោយរលូន។

- ឯកឧត្តមបណ្ឌិត ស្រ៊ីវ ថារិទ្ធ អ្នកសម្របសម្រួលសាលាក្រោយឧត្តម និង លោកបណ្ឌិត សុខ សិរី អនុប្រធានការិយាល័យស្រាវជ្រាវ ដែលតែងជួយឈឺឆ្អាល ជួយសម្របសម្រួល និងផ្តល់សេវាសម្រាប់ការប្រជុំបច្ចេកទេសនានាតាមអនឡាញតាំងពីដើមរៀងមក។

- អស់លោក-លោកស្រី ជាសមាជិក-សមាជិកាគណៈកម្មការអភិវឌ្ឍសៀវភៅសិក្សា ក៏ដូចជាគណៈកម្មការត្រួតពិនិត្យសៀវភៅសិក្សារបស់សាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ ដែលបានជួយផ្តល់ធាតុចូលដល់សៀវភៅនេះឲ្យកាន់តែមានគុណភាពប្រសើរថែមទៀត។

- ជាចុងក្រោយ និងជាពិសេស លោក ធូ វណ្ណា ប្រធានការិយាល័យស្រាវជ្រាវនៃសាកលវិទ្យាល័យភូមិន្ទភ្នំពេញ ដែលតែងជួយផ្តល់ធាតុចូលចំពោះរាល់ឯកសារការងារ ទាំងផ្នែករដ្ឋបាលនិងទាំងផ្នែកបច្ចេកទេស លោកថែមទាំងជួយសម្រួលរាល់បញ្ហាប្រឈមនានាយ៉ាងដឹតដល់បំផុត ដោយមិនប្រកាន់ពេលវេលា និងដោយចិត្តជ្រះថ្លា រហូតការសរសេរសៀវភៅនេះទទួលបានជោគជ័យជាស្តាពរ។

មាតិកា

ជំពូកទី១

មូលដ្ឋានគ្រឹះរបស់ Network

១-១-តើ ណេតវើកគឺជាអ្វី ?

ណេតវើកួមមានកុំព្យូទ័រពីរឬច្រើនត្រូវបានគេភ្ជាប់គ្នាក្នុងគោលបំណងចែកចាយការប្រើធនធានដូចជា Printer, CD-ROMs និងផ្លាស់ប្តូរ files ឬអនុញ្ញាតឲ្យវាធ្វើការទំនាក់ទំនងតាមអេឡិចត្រូនិច។ កុំព្យូទ័រនៅលើបណ្តាញណេតវើកត្រូវបានភ្ជាប់គ្នាតាមរយៈខ្សែ ណេតវើកប្រព័ន្ធទូរស័ព្ទ រលកវិទ្យុ ផ្កាយរណបឬពន្លឺ infrare។

១-២-ប្រភេទមូលដ្ឋានគ្រឹះនៃ ណេតវើក

- Personal Area ណេតវើក(PAN)
- Local Area ណេតវើក(LAN)
- Wide Area ណេតវើក(WAN)
- Metropolitan Area ណេតវើក(MAN)

១-២-១-PAN (Personal Area Network)

វាគឺជាប្រភេទ ណេតវើកដែលទើបតែចេញឲ្យប្រើប្រាស់សម្រាប់អ្នកជំនួញដែលងាយស្រួលក្នុងការបញ្ជូនព័ត៌មានឬ Fileពីមនុស្សម្នាក់មកមនុស្សម្នាក់ទៀតដូចជា BlueTooth, Infrared ឬ Wifi ជាដើម ។

១-២-២-Local Area Network

Local Area ណេតវើក(LAN) វាគឺជាប្រភេទនៃ ណេតវើកមួយដែលត្រូវបានបង្កើតឡើងសម្រាប់ប្រើប្រាស់នៅតាមតំបន់តូចៗដូចជា ប្រើប្រាស់សម្រាប់បន្ទប់ពិសោធន៍ សាលារៀននិងអាគារជាដើម។

ក្នុងការដំឡើង LAN មួយគេត្រូវការកុំព្យូទ័រមួយប្រើជា Fileserver។ វាមានតួនាទីសម្រាប់ផ្ទុកនូវរាល់Softwareទាំងអស់ឲ្យប្រព័ន្ធ Networkទាំងមូលប្រើនិងសម្រាប់គ្រប់គ្រង់លើបណ្តាញ ណេតវើកនិងចែកចាយឲ្យClients ដែលស្ថិតនៅក្នុងប្រព័ន្ធ ណេតវើកនោះប្រើប្រាស់។

កុំព្យូទ័រដែលភ្ជាប់មកកាន់ Fileserver ហៅថា Workstation។ Workstations គ្មានសមត្ថភាពដូចFileserver នោះទេហើយគេអាច Install software មួយចំនួនលើវាបាន។ LAN ភាគច្រើនត្រូវបានគេភ្ជាប់វាដោយប្រើ NIC (ណេតវើកInterface Card) ។

១-២-៣-Wide Area Network

Wide Area ណេតវើក(WANs) គឺជាប្រព័ន្ធ ណេតវើកមួយដែលត្រូវបានគេប្រើសម្រាប់ភ្ជាប់តាមតំបន់ដែលមានភូមិសាស្ត្រធំៗដូចជានៅរដ្ឋ Florida, United States ជាដើម។ គេអាចភ្ជាប់វាតាមខ្សែដែលឆ្លងកាត់មហាសមុទ្រឬតាមផ្កាយរណបជាដើម។

បើនិយាយអំពីការប្រើ WAN សាលារៀនជាច្រើននៅរដ្ឋ Florida អាចភ្ជាប់ជាមួយសាលារៀននៅទីក្រុងតូក្យូនៃប្រទេសជប៉ុនដោយចំណាយតិចបំផុត។

WAN មានលក្ខណៈសាំញ៉ាំពីព្រោះវាប្រើ Multiplexers ដើម្បីភ្ជាប់ Local និងMAN (Metropolitan Area Network) ទៅកាន់ពិភពលោកដូចជា Internet ។

គុណសម្បត្តិនៃការប្រើ ណេតវើកនៅតាមសាលារៀន

- **Speed:** វាងាយស្រួលក្នុងការចែកចាយឯកសារ។ បើគ្មាន ណេតវើកទេ យើងត្រូវថតចំលងដាក់ Floppy ឬ Flash ហើយធ្វើការចំលងពីកុំព្យូទ័រមួយដាក់ក្នុងកុំព្យូទ័រមួយទៀត។
- **Cost:** ចំណេញក្នុងការទិញ Software សព្វថ្ងៃដែលមានអាជ្ញាប័ណ្ណ។ ក្រៅពីការសន្សំសំចៃប្រាក់ វាអាចចែកចាយ Software មានល្អប្រសើរទៀត។
- **Security:** គ្រប់ Software ឬ Data ត្រូវបានរក្សាទុកនៅលើ Server ។ ដូច្នេះគេមិនអាចលួចចំលងបាននោះទេ បើគ្មានសិទ្ធិ។គ្រប់ទិន្នន័យទាំងអស់ត្រូវបានការពារដោយប្រព័ន្ធ Security របស់ ណេតវើក។
- **Centralized Software Management:** រាល់ Application Servers ទាំងអស់ត្រូវបានដំឡើងនៅលើ Server ។ ដូច្នេះយើងអាចគ្រប់គ្រង់បានទាំងអស់ក្នុងការចែកចាយឯសារឬផ្តល់សេវា។
- **ធនធាន Sharing:** យើងអាចចែកចាយ ធនធានs ដែល Server មានទៅឲ្យគ្រប់ Clients ដែលគ្មាន Resourses ទាំងនោះដូចជា CD-ROM,Floppy,Flash និងprinter Laser ដែលមានតម្លៃថ្លៃជាដើម។ ជាពិសេសទៅទៀតនោះវាមានកាត់បន្ថយក្នុងការទិញ Software ឬ hardware ដែលមានតម្លៃខ្ពស់។
- **Electronic Mail:** គឺមធ្យោបាយមួយដ៏ល្អប្រសើរបំផុតដែលមនុស្សបច្ចុប្បន្នប្រើប្រាស់សម្រាប់ផ្តល់ព័ត៌ មានឲ្យគ្នាទៅវិញទៅមកបានយ៉ាងឆាប់រហ័សពាសពេញពិភពលោក។
- **Flexible Access:** ជួយសំរួលដល់និស្សិតក្នុងការ Access យកនៅ Assignment ដែលបានដាក់នៅលើប្រព័ន្ធណេតវើកបានយ៉ាងឆាប់រហ័សហើយអាចធ្វើការរួមគ្នាពីផ្ទះរបស់គេបាន។
- **Workgroup Computing:** Workgroup software ដូចជា Microsoft BackOffice អនុញ្ញាតឲ្យអ្នកប្រើប្រាស់ធ្វើការលើ Document ឬគម្រោងព្រមគ្នា។

ឧទាហរណ៍

អ្នកសិក្សានៅក្នុងសាលានីមួយៗនៅគ្រប់ទីកន្លែងទាំងអស់អាចចូលរួមក្នុងការបញ្ចេញមតិទៅលើកម្មវិធីសិក្សាឬឯកសារផ្សេងដែលគេមិនពេញចិត្ត។

គុណវិបត្តិនៃការដំឡើងប្រព័ន្ធ ណេតវើកនៅតាមសាលា

- **Expensive to Install:** យើងត្រូវចំណាយទុនច្រើនទៅលើការទិញសំភារៈ ដូចជា ខ្សែ ណេតវើកឧបករណ៍ Networks និងត្រូវការចំណាយទៅលើអ្នកបច្ចេកទេសក្នុងដំឡើងវា។
- **Requires Administrative Time:** អ្នកជំនាញត្រូវតែច្បាស់លាស់ក្នុងការមើលថែទាំវា និងគ្រប់គ្រង់លើវា។ ជាពិសេសធានានិរន្តភាពរបស់វា។
- **File Server May Fail:** ប្រព័ន្ធ ណេតវើកពឹងផ្អែកទាំងស្រុងទៅលើ Server ។ បើ Server ខូចនោះប្រព័ន្ធទាំងមូលត្រូវគាំងទាំងអស់។
- **Cable May Break:** បើខ្សែសំខាន់ត្រូវបានដាច់ នោះវាបានធ្វើឲ្យអាក់ខានក្នុងការបញ្ជូនទិន្នន័យឬអាចបាត់បង់ទិន្នន័យក្នុងកំឡុងពេលនៃការបញ្ជូន។

សំណួរត្រួតពិនិត្យ

១-តើ ណេតវើកគេចែកចេញជាប៉ុន្មានប្រភេទសព្វថ្ងៃនេះ?

២-តើ ណេតវើកផ្តល់នូវគុណសម្បត្តិនិងគុណវិបត្តិអ្វីខ្លះ?

៣-តាមយោបល់របស់អ្នក តើអ្នកគិតថានៅប្រទេសយើងត្រូវភ្ជាប់ប្រព័ន្ធ ណេតវើកនៅតាមសាលារៀនឬទេ ?

១-៣-តើអ្វីទៅជា Protocol ?

Protocol គឺជាសំនុំនៃក្បួនច្បាប់ដែលគេប្រើសម្រាប់គ្រប់គ្រងទៅលើទំនាក់ទំនងរវាងកុំព្យូទ័រពីរឬច្រើននៅលើបណ្តាញណេតវើក។ ច្បាប់ទាំងនោះរួមមានការណែនាំដែលតម្រូវឲ្យប្រើប្រាស់នូវណេតវើកឲ្យបានត្រឹមត្រូវគឺ Access Method ដែលបានអនុញ្ញាតឲ្យរួបរាងនៃណេតវើក (Topology)និងល្បឿននៃការបញ្ជូនទិន្នន័យ។ Protocol ដែលគេនិយមប្រើភាគច្រើនបំផុតមានដូចជា:

- Ethernet
- LocalTalk
- Token Ring
- FDDI
- ATM

១-៣-១-Ethernet

Ethernet គឺជាប្រភេទនៃ ណេតវើកមួយដែលមានការពេញនិយមប្រើប្រាស់ក្នុងប្រភេទនៃ ណេតវើកមានទំហំធំ។ វាប្រើ Access Method មានឈ្មោះថា **CSMA/CD** ។ នេះគឺជាប្រភេទនៃ system មួយដែលគ្រប់ Devices ទាំងអស់ត្រូវស្តាប់ខ្សែ ណេតវើកជាមុនសិនទើបអាចធ្វើការបញ្ជូនទិន្នន័យបាន។ បើមានឧបករណ៍មួយកំពុងបញ្ជូននោះវាត្រូវរង់ចាំរហូតទាល់តែមានបញ្ជូនរួចឬទំនេរ។ បើមាន Stations ពីរធ្វើការបញ្ជូននៅពេលព្រមគ្នាតាមទិសដៅផ្ទុយគ្នានោះនិងមាន collision កើតឡើងជាពុំខាន។ បើមាន collision កើតឡើងនោះវានិងកាត់បន្ថយប្រសិទ្ធិភាពនៃការបញ្ជូនទិន្នន័យ ។

១-៣-២-Fast Ethernet

គឺជាប្រភេទនៃ Ethernet ដែលបានមកពីការបន្ថែមល្បឿនឲ្យវាមានល្បឿនរហូតដល់១០០Mbps។ វាប្រើប្រាស់ខ្សែ cat5 ដែលមានល្បឿនលឿននិង concentrator ដែលមានល្បឿនលឿនដែរ។ សព្វថ្ងៃនេះត្រូវបានគេពេញនិយមប្រើប្រាស់វាណាស់នៅតាមសាលារៀន។

១-៣-៣-Gigabit Ethernet

មានការបង្កើតថ្មីៗនៅក្នុងស្តង់ដានៃ Ethernet គឺជា Protocol មួយដែលមានអត្រានៃការបញ្ជូនមានល្បឿន 1Gbps ។ Gigabit Ethernet ត្រូវបានគេប្រើសម្រាប់ Backbone ណេតវើកនៅពេលនេះ។ នៅពេលអនាគត់វាប្រហែលជាត្រូវបានគេប្រើសម្រាប់ Workstation និង Server ផងដែរ។ វាអាចជា Fiber Optic និង Copper ផងដែរ។ 1000 Base Tx cable ត្រូវបានប្រើសម្រាប់ Giggabit Ethernet ហើយវាបានក្លាយជាស្តង់ដាជាផ្លូវការនៅឆ្នាំ១៩៩០។

១-៣-៤-LocalTalk

LocalTalk គឺជា ណេតវើកprotocol មួយដែលត្រូវបានបង្កើតឡើងដោយក្រុមហ៊ុន Apple កុំព្យូទ័រនិង Macintosh។ Access Method ដែលប្រើដោយ LocalTalk មានឈ្មោះថា CSMA /CA(Carrier Sense Multiple Access with collision Avoidance) ។ វាស្រដៀងទៅនិង CSMA/CD ដែរលើកលែងតែសញ្ញាល់របស់កុំព្យូទ័រគឺវាបានប្រាប់មុនឲ្យដឹងមុនពេលវាធ្វើការ។ localTalk និងខ្សែ UTP ត្រូវបានគេប្រើសម្រាប់ភ្ជាប់ពីកុំព្យូទ័រតាមរយៈ Serial Port។ ចំពោះប្រព័ន្ធប្រតិបត្តិការនៃ Macintosh អនុញ្ញាតឲ្យបង្កើតការភ្ជាប់ជា Peer-to-Peer ដោយមិនចាំបាច់បន្ថែមនូវ Software នោះទេ។

ជាមួយការបន្ថែមនូវ Server Version នៃប្រភេទ Client/Server Software ដែលត្រូវបានបង្កើតឡើង។ LocalTalk Protocol អនុញ្ញាតឲ្យរូបរាងនៃណេតវើកមានទម្រង់ជា Linear bus star ឬជា Tree ដោយប្រើ UTP។ គុណសម្បត្តិរបស់វាគឺយឺត។ ល្បឿននៃការបញ្ជូនរបស់វាគឺត្រឹមតែ២៣០ Kbps ។

១-៣-៥-Token Ring

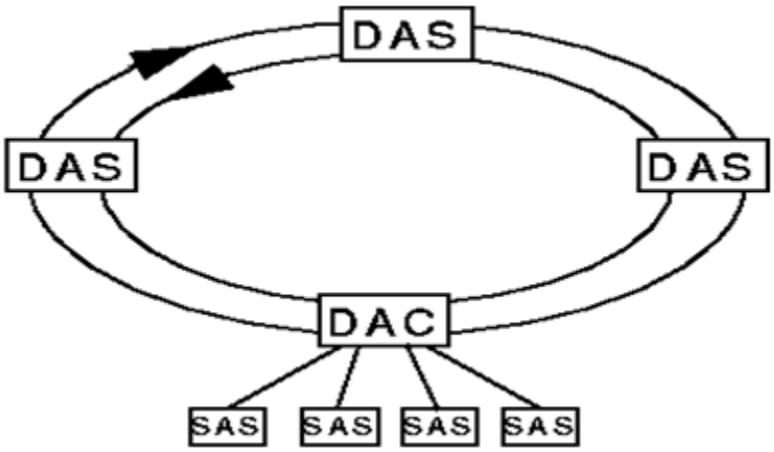
Token Ring protocol ត្រូវបានបង្កើតឡើងដោយក្រុមហ៊ុន IBM ក្នុងឆ្នាំ១៩៨០។ Access Method របស់វាគឺជា Token Passing។ នៅក្នុង Token passing គឺកុំព្យូទ័រទាំងអស់ត្រូវបានភ្ជាប់ជាមួយគ្នាជារង្វង់។ ដូច្នេះសញ្ញាល់របស់វាធ្វើដំណើរជុំវិញណេ ណេតវើកពីកុំព្យូទ័រមួយទៅកុំព្យូទ័រមួយទៀតក្នុងទម្រង់ជារង្វិលជុំ។ អេឡិចត្រូនិកតែមួយគត់បំលាស់ទីជុំវិញនៃ Ring ពីកុំព្យូទ័រមួយទៅកុំព្យូទ័រមួយទៀត។ ប្រសិនបើកុំព្យូទ័រមួយគ្មានព័ត៌មានត្រូវបញ្ជូននោះទេ វាបញ្ជូន Token ទៅឲ្យកុំព្យូទ័របន្ទាប់។ បើកុំព្យូទ័រមួយមានបំណងចង់ធ្វើការបញ្ជូននឹងទទួលនូវ Token ទទេរមួយ វាភ្ជាប់ទិន្នន័យទៅកាន់ Token។ Token បំលាស់ទីជុំវិញ Ring រហូតមកដល់កុំព្យូទ័រដែលមានផ្ទុកទិន្នន័យ។ ចំណុចនេះទិន្នន័យត្រូវបានចាប់យកដោយកុំព្យូទ័រដែលជាអ្នកទទួល។ Token Ring Protocol ត្រូវការ Star-Wired Ring ដោយប្រើ UTP ឬជា Fiber Optic ។ វាអាចធ្វើការបញ្ជូនដែលមានល្បឿន 4Mbps ឬ 16 Mpbs ។ ដោយសារតែមានការពេញនិយមប្រើ Ethernet ធ្វើឲ្យ Token Ring មានការធ្លាក់ចុះ។

១-៣-៦-FDDI

FDDI (Fiber Distributed Data Interface) គឺជា ណេតវើកProtocol មួយដែលត្រូវបានបង្កើតឡើងសម្រាប់ភ្ជាប់ Local Area ណេតវើកតាមធម្មតាដែលមានរយៈចម្ងាយធ្ងាយ។ Access Method របស់វាគឺជាប្រើ Token Passing ។ FDDI ប្រើរូបរាងដែលមានខ្សែពីរ។ ការបញ្ជូនទិន្នន័យតាមធម្មតាប្រើខ្សែតែមួយក្នុងចំណោមខ្សែជាច្រើននោះ។ ប្រសិនបើមានការខូចខាតកើតមានឡើងចំពោះប្រព័ន្ធនោះព័ត៌មានត្រូវបានបញ្ជូនដោយប្រើខ្សែទី២ក្នុងការបង្កើតឡើងនូវ Ring ថ្មី។ គុណសម្បត្តិនៃ FDDI គឺវាមានល្បឿនលឿន។ វាប្រតិបត្តិការលើខ្សែ Optic មានល្បឿន 100Mpbs ។

ណេតវើកConfiguration

FDDI - ណេតវើកSchematic



DAS : Dual Attach Station

DAC: Dual Attach Concentrator

SAS:Single Attach Station

FDDI - 2 Counter Rotating Rings

រូបភាព១-បង្ហាញពី Ring ដែលមានខ្សែពីរនិងឧបករណ៍ប្រើជាមួយវា

- Rings ទាំងពីររបស់ fibre ត្រូវបានដំឡើងគឺ - Primary and Secondary ring
- Token និង data flow តាមទិសដៅពីរផ្សេងគ្នាក្នុង Ring នីមួយៗ
- Ring ទី២ប្រហែលត្រូវបានប្រើប្រាស់ជាខ្សែបម្រុង (Backup) នៅពេលដែល Ring ទី១ត្រូវបានខូច
- Stations នៅក្នុង FDDI Ring ត្រូវបានភ្ជាប់ទៅ Ring ទាំងពីរ

Fault tolerance

- Fibre break – Port ពីរជាប់គ្នាខូចវាត្រូវបានផ្តាច់ហើយ Stations ទាំងនោះត្រូវបានមិនដំណើរការ មានន័យថា Ring ក្លាយជា Ring តែមួយ។
- Station failure–Stations ពីរនៅជិតបំផុតត្រូវបានខូច មានន័យថា Ring ក្លាយជា Ring តែមួយ
- Multiple breaks - rings ត្រូវបានដាច់ជាកង់ៗ

របៀបនៃការភ្ជាប់

Workstations

- Device ដែលមាន port ពីរត្រូវបានភ្ជាប់ដោយផ្ទាល់ជាមួយ Ring ទាំងពីរដែលគេហៅថា DAS (Dual Attachment Station) ។ គេកំភ្ជាប់វាជាមួយ workstations ណាស់នៅលើ FDDI backbone
- SAS - Single Attachment Station -ត្រូវបានគេភ្ជាប់ជាមួយ primary ring

Concentrator

- DAC Dual Attachment Concentrator - SASs in a star
  - គេនិយមប្រើវា
    - Fault tolerant eg. powering down workstation
    - សម្រាប់ផ្តល់ជាគម្រោងល្អសម្រាប់ការដំឡើង network
    - ចំណាយតិចក្នុងការភ្ជាប់
    - មិនរំខានដល់ការភ្ជាប់
    - ងាយស្រួលដល់ការដោះស្រាយរាល់បញ្ហាដែលបានកើតឡើង
  - គេមិននិយមប្រើ

- បើចំណុចមួយខូច វាធ្វើឲ្យ workstations ជាច្រើនខូចទាំងអស់
- Concentrators អាចរៀបតាមលំដាប់ថ្នាក់បង្កើតបានជា Tree
- SAC - Single Attachment Concentrator - សម្រាប់ភ្ជាប់មកកាន់ DAC type concentrator.
- Dual homing - គឺជា DAS មួយត្រូវបានគេភ្ជាប់ជាមួយ concentrator ports ដែល port A គឺជា hot standby សម្រាប់ជាជំនួយឲ្យ B port ក្នុងករណីវាខូច

ប្រភេទនៃ Port

- មាន Port 4 ប្រភេទខុសគ្នាគឺ A, B, S, M
- DASs មាន 2 ports A និង B ដែលត្រូវបានភ្ជាប់ឆ្លាស់គ្នា មានន័យថា port A នៃ DAS គឺត្រូវបានភ្ជាប់ជាមួយ port B នៃ DAS មួយទៀត
- SASs មាន S port តែមួយគត់
- Concentrators មាន M ports ជាច្រើនដែលអាចភ្ជាប់ជាមួយ S ports នៃ SASs (ឬ A and B ports of DASs)

Port connections

- ការភ្ជាប់ខុសបច្ចេកទេស

M - M

- ពេញនិយមភ្ជាប់

A - B A - M B - M M - S

- ការភ្ជាប់ដែលបណ្តាលឲ្យមានបញ្ហា (Vendor dependent)

A - A B - B S - A S - B S - S
ខ្សែ Fiber Optic

- Multimode fibre 62.5/125 micron (fibre/casing)
- អាចប្រើជាមួយ 50/125, 85/125 ...
- Single mode 8-12 micron
- 62.5/125 សម្រាប់ LED/photodiode technology ត្រូវបានប្រើប្រាស់សម្រាប់រកឲ្យឃើញពន្លឺនៅក្នុងខ្សែដែលត្រូវការ fibre ទំហំធំ
- 50/125 ប្រើប្រាស់ដោយក្រុមហ៊ុន PTTs ជាច្រើនក្នុងប្រទេសអឺរ៉ុប
- 8-12-micron cable ត្រូវបានប្រើប្រាស់ជាមួយ Laser

Cabling

- 4 fibres (2 transmit and 2 receive)
- ដំឡើង of spares recommended for replacement of faulty fibres

FDDI over Copper – CuDDI

- ប្រើ copper UTP (Unshielded Twisted Pair) ឬ STP (Shielded Twisted Pair) cables សម្រាប់ desktop FDDI
- មានប្រវែងអតិបរមាន ១០០ម៉ែត្រ
- Standard: ANSI TP-PMD (Twisted Pair – Physical Medium Dependent)
- ខ្សែមានតម្លៃថោក ងាយស្រួលក្នុងការដំឡើងនិងងាយស្រួលក្នុងការភ្ជាប់
- Copper transceivers មានតម្លៃថោក តូចជាងនិងត្រូវការថាមពលតិច

១-៣-៧-ATM

ATM (Asynchronous Transfer Mode) គឺជា ណេតវើកprotocol មួយដែលបញ្ជូនទិន្នន័យនូវអត្រាល្បឿន 155 Mbps ឬខ្ពស់ជាងនេះ។ ATM ធ្វើការដោយបញ្ជូនទិន្នន័យទាំងអស់នៅក្នុងទម្រង់ជា Packets តូចៗនៃទំហំថេរពីព្រោះថា Protocol ផ្សេងទៀតបញ្ជូន Packets មានប្រវែងប្រែប្រួល។ ATM អាចប្រើជាមួយ Media មានដូចជា Video, CD-quality audio និង imaging។ ATM ធ្វើការជាមួយរាងជា STAR ដែលអាចធ្វើជាមួយ Fiber Optic និង UTP ផងដែរ។

ATM ភាគច្រើនត្រូវបានភ្ជាប់ជា LAN ពីរឬច្រើន។ វាត្រូវបានប្រើដោយ IPS (Internet Service Provider) ផងដែរដើម្បីដំណើរការដែលមានល្បឿនលឿនទៅកាន់ Internet សម្រាប់អតិថិជនរបស់គេ។ បច្ចេកវិទ្យា ATM កំពុងពេញនិយមដែលផ្តល់ជាដំណោះស្រាយផ្សេងៗសម្រាប់ធ្វើឲ្យ LAN ល្បឿនលឿន។

តារាងសង្ខេបពី Protocol

Protocol	Cable	Speed	Topology
Ethernet	Twisted Pair, Coaxial, Fiber	10 Mbps	Linear Bus, Star, Tree
Fast Ethernet	Twisted Pair, Fiber	100 Mbps	Star
LocalTalk	Twisted Pair	.23 Mbps	Linear Bus or Star
Token Ring	Twisted Pair	4 Mbps - 16 Mbps	Star-Wired Ring
FDDI	Fiber	100 Mbps	Dual ring
ATM	Twisted Pair, Fiber	155-2488 Mbps	Linear Bus, Star, Tree

រូបភាព២-បង្ហាញពីតារាងសង្ខេបនៃ ណេតវើកProtocols

សំណួរត្រួតពិនិត្យ

១-តើ protocol ជាអ្វី?

២-តើ Protocol ដែលអ្នកបានសិក្សាមានអ្វីខ្លះ?ចែកចេញជាប៉ុន្មានប្រភេទ?

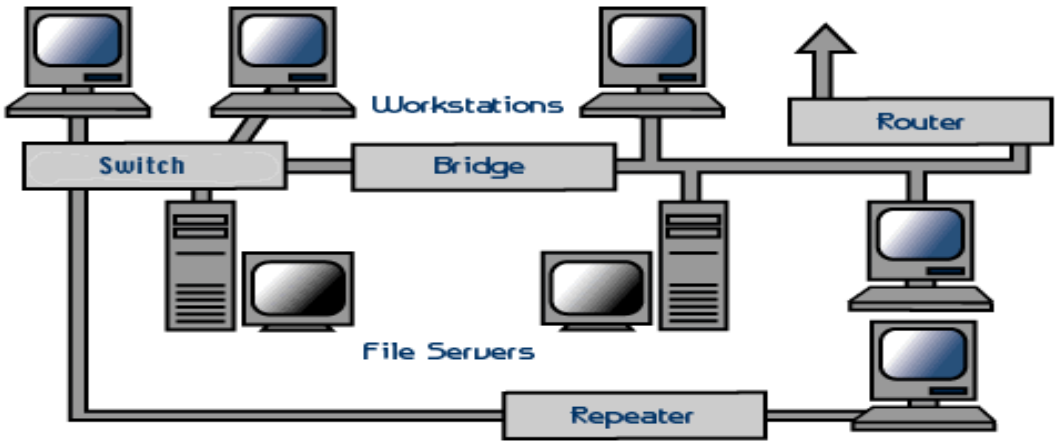
៣-តើ Protocol ណាដែលគេនិយមប្រើជាងគេ?ហេតុអ្វី?

៤-តើ FDDI ខុសពី Protocol ផ្សេងទៀតយ៉ាងដូចម្តេច?

៥-តើចំណុចពិសេសរបស់ FDDI ត្រង់ចំណុចណា?

១-៤-តើ Networking Hardware ជាអ្វី?

Networking hardware រួមមានកុំព្យូទ័រ, peripherals, Interface cards និងឧបករណ៍ផ្សេងៗទៀត សម្រាប់ដំណើការទិន្នន័យនិងធ្វើការទំនាក់ទំនងនៅលើណេតវើក។ សូមពិនិត្យទៅលើរូបភាពខាងក្រោមស្តីអំពីណេតវើកhardware ទាំងនោះ។



រូបភាព៣-បង្ហាញពីការដំឡើង ណេតវើកដោយប្រើ Networking devices

ផ្នែកនេះផ្តល់ព័ត៌មានអំពីសមាសភាពដូចខាងក្រោម:

- File Servers
- Workstation
- ណេតវើកInterface Cards(NIC)
- Switches
- Repeater
- Bridges
- Router

១-៤-១-File Servers

Fileserver គឺជាផ្នែកមួយដ៏សំខាន់របស់ប្រព័ន្ធ Network។ វាគឺជាកុំព្យូទ័រមេមួយដែលមានទំហំ RAM ធំ មាន Hard disk ទំហំធំនិងមានភ្ជាប់មកជាមួយនូវ ណេតវើកCard ដែលមានល្បឿនលឿន។ ណេតវើកOperating System (NOS)ក៏ត្រូវបានគេដំឡើងរួចជាស្រេចនៅក្នុងកុំព្យូទ័រនោះផងដែរ។ ក្រៅពីនេះទៀតគេបានដំឡើងនូវ Software application និងទិន្នន័យមួយចំនួនទៀតត្រូវបានគេប្រើដើម្បីចែកចាយ។

Fileserver ត្រូវបានគេប្រើប្រាស់ដើម្បីគ្រប់គ្រងទៅលើប្រព័ន្ធណេតវើកដែលអាចឲ្យវាបញ្ជូននូវ Word processor ទៅឲ្យ Workstation មួយផ្សេងទៀតហើយរក្សាទុកនូវ Email message ជាមួយវាងដែរ។ នេះគឺជាតម្រូវការនៃកុំ

ព្យូទ័រមួយដែលអាចផ្ទុកព័ត៌មានជាច្រើនអាចចែកចាយបានក្នុងល្បឿនយ៉ាងលឿន។ Fileserver យ៉ាងហោចណាស់ត្រូវមានលក្ខណៈដូចខាងក្រោម:

- 800 megahertz or microprocessor ឬលឿនជាងនេះ (Pentium 3 or 4, G4 or G5)
- Fast hard drive យ៉ាងតិចបំផុតផ្ទុកបាន 120 gigabytes
- RAID (Redundant Array of Inexpensive Disks) ដើម្បីការបាត់បង់ទិន្នន័យបន្ទាប់ពីមាន Disk ណាមួយខូច

SCSI Cabling

-Internal Devices

- 68 pin cable

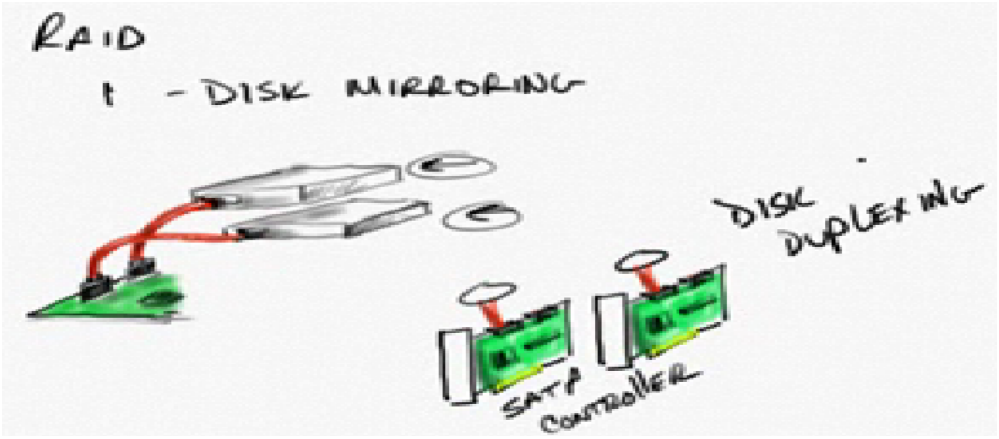
-External devices

- 50 pin or 68 pin

SCSI(Small កុំព្យូទ័រ Systems Interfaces

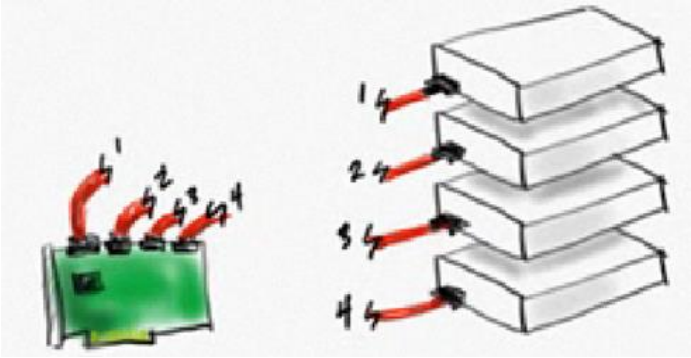
- 1970's

RAID-1:Mirroring



RAID-5:Disk Stripe Set with Parity

Disk Striping with Parity



RAID

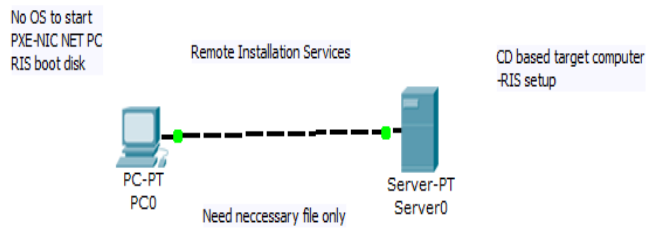
- Seril ATA
- Parallel ATA

Hardware vs. Software Configuration

Server អាចដើរតួនាទីជា RIS Server

RIs server ត្រូវការ DAD

- DNS-IP
- Active Directory ដើម្បីរក្សាទុក RIS Server
- DHCP-សម្រាប់ផ្តល់IP addresses ជាស្វ័យប្រវត្តិ



- A tape backup-up unit (i.e. DATA, JAZ, Zip, ឬ CD-RW drive) គឺជាឧបករណ៍សម្រាប់រក្សាទុកទិន្នន័យនៅពេលធ្វើការ backup
- មាន expansion slot ជាច្រើន
- ណេតវើកinterface card មានល្បឿនលឿន
- RAM យ៉ាងហោចណាស់ 512 MB

១-៤-២-Workstations

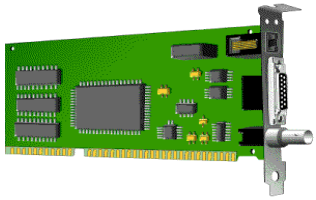
គ្រប់កុំព្យូទ័រទាំងអស់ដែលភ្ជាប់ទៅកាន់ណេតវើកគេហៅថា Workstation។ Workstation គឺជាកុំព្យូទ័រមួយដែលត្រូវបានដំឡើងជាមួយ ណេតវើកInterface card ណេតវើកsoftware និងខ្សែសមរម្យ។ Workstation មិនចាំបាច់ត្រូវការនូវ Flopyy Disk drive នោះទេពីព្រោះថា Files អាចត្រូវបានរក្សាទុកនៅលើ Fileserver នោះ។ គ្រប់កុំព្យូទ័រទាំងអស់អាចប្រើជា ណេតវើកstation មួយបាន។

១-៤-៣-ណេតវើកInterface Cards

ណេតវើកinterface card(NIC)សម្រាប់ផ្តល់ឲ្យនូវការភ្ជាប់រវាង ណេតវើកនិង Copmuter Workstation។ NICs ភាគច្រើនគឺជា Card ដែលស្ថិតនៅខាងក្នុងដែលដោតភ្ជាប់ទៅនិង expansion slot នៅខាងក្នុងនៃកុំព្យូទ័រ។ កុំព្យូទ័រមួយចំនួនគឺជា Macintosh ដែលប្រើប្រអប់ខាងក្រៅដែលអាចភ្ជាប់ជាមួយ Serial port ឬ SCSI port ។ កុំព្យូទ័រ Laptop ឥឡូវអាចទិញភ្ជាបមកជាមួយនូវ ណេតវើកcard ឬជាមួយ ណេតវើកcard អាចជា PCMCIA slot ។

ណេតវើកInterface Cards គឺជាកត្តាចំបងជាងគេក្នុងការកំនត់ល្បឿននិងការបញ្ជូនទិន្នន័យក្នុងណេតវើក។ វាគឺជាគំនិតល្អដែលប្រើ ណេតវើកcards ដែលមានល្បឿនលឿនសម្រាប់ Workstation ។

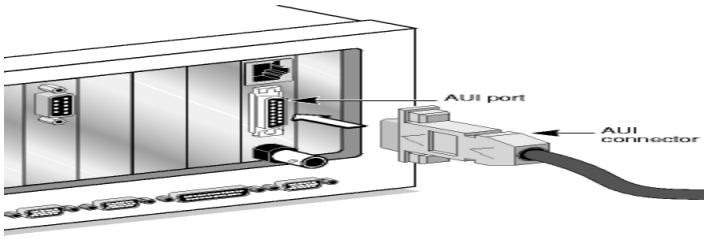
ណេតវើកinterface Card មានបីប្រភេទគឺ Ethernet cards, LocalTalk connectors និង Token Ring cards ។



១-៤-៤-Ethernet Cards

Ethernet cards ជាធម្មតាទិញមកជាច់ដោយឡែកពីកុំព្យូទ័រ ទោះបីជាកុំព្យូទ័រជាច្រើនដូចជា Macintosh មានភ្ជាប់ជាមួយ option សម្រាប់ភ្ជាប់ Enternet card ក៏ដោយ។ Ethernet card កន្លែងសម្រាប់ប្រើទាំងខ្សែជា Caxial និង UTP។

ប្រសិនជាវាត្រូវបានបង្កើតឡើងសម្រាប់ខ្សែCoaxial នោះវាជាប្រភេទ BNC។ ប្រសិនបើវាត្រូវបានបង្កើតឡើងសម្រាប់ UTP នោះវាជា RJ-45។ Ethernet card មួយចំនួនមានប្រភេទជា AUI ផងដែរ។



LocalTalk Connectors

LocalTalk គឺជាផលិតផលរបស់ក្រុមហ៊ុន Apple សម្រាប់កុំព្យូទ័រ Macintosh networking។ វាប្រើប្រាស់ Adapter ពិសេសនិងខ្សែដែលភ្ជាប់ port printer នៃ Macintosh។ គុណវិបត្តិនៃ LocalTalk គឺថាវាមានល្បឿនយឺតបើប្រៀបធៀបជាមួយ Ethernet។ Ethernet ភាគច្រើនប្រតិបត្តិបាន 10Mbps។ ផ្ទុយមកវិញ LocalTalk ប្រតិបត្តិការបានត្រឹមតែ 230 Kbps ឬ .23 Mbps ។

Ethernet Cards ប្រៀបធៀបជាមួយ LocalTalk

Ethernet	LocalTalk
មានល្បឿនលឿនក្នុងការបញ្ជូនទិន្នន័យ(ពី១០ដល់ ១០០ Mbps)	មានល្បឿនយឺតក្នុងការបញ្ជូនទិន្នន័យ(.២៣ Mbps)
មានតម្លៃថ្លៃពីព្រោះទិញដាច់ដោយឡែកពីគ្នា	មានស្រាប់នៅក្នុង Macintosh
ត្រូវការ Slot	មិនត្រូវការ Slot
អាចប្រើបានជាមួយកុំព្យូទ័រភាគច្រើន	ប្រើបានតែជាមួយ Macintosh

រូបភាព៤-បង្ហាញពីការប្រៀបធៀប Ethernet និង LocalTalk network

Token Ring Cards

Token Ring ណេតវើកcards មានលក្ខណៈស្រដៀងទៅនិង Ethernet card ដែរ។ គេមើលឃើញភាពខុសគ្នាប្លែកមួយគឺប្រភេទនៃ Connector នៅខាងក្រោយនៃ Card ។ Token Ring Card ជាទូទៅមាន៩ម្ជុល DIN ប្រភេទ Connector ដើម្បីភ្ជាប់ Card ទៅនិងខ្សែ ណេតវើក។

១-៤-៥-Switch

Concentrator គឺជាឧបករណ៍មួយដែលផ្តល់ឲ្យនូវចំណុចតភ្ជាប់កណ្តាលសម្រាប់ខ្សែពី Workstation Server និង Pheripherals។ នៅក្នុងរូបរាងនៃ Star ខ្សែ UTP ត្រូវបានភ្ជាប់ពីគ្រប់ Workstation ទៅកាន់ Switch /Hub។ Switch ភាគច្រើនគឺជា Active ដែលវាមានកំលាំងសញ្ញាដែលអាចបំលាស់ទីពីឧបករណ៍មួយទៅឧបករណ៍មួយទៀត។ Switch មិន Broadcast ដូច Hub នោះទេ។ វាចាំនូវ Address របស់កុំព្យូទ័រនិងបញ្ជូនព័ត៌មានទៅកាន់គោលដៅបានច្បាស់លាស់។

Swichs គឺ

- ប្រើ configured with 8, 12, ឬ 24 RJ-45 ports
- ប្រើនៅក្នុង star ឬ star-wired ring topology
- មាន software ពិសេសសម្រាប់គ្រប់គ្រងទៅលើ port
- ជាធម្មតាជំទ្បើងនៅក្នុងស្តង់ដាជាលោហៈធាតុដែលអាចផ្ទុកនូវ netmodems, bridge ឬ routers

១-៤-៦-Repeaters

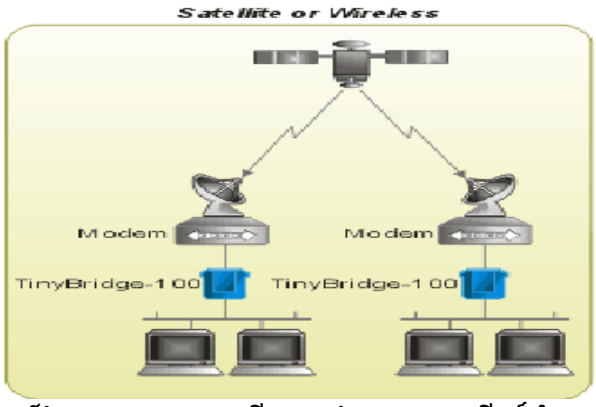
ដោយសារតែសញ្ញាលចុះខ្សោយនៅពេលដែលឆ្លងកាត់ខ្សែ ដូច្នេះគេត្រូវការធ្វើឲ្យសញ្ញានោះខ្លាំងឡើងវិញ គេប្រើឧបករណ៍មួយហៅថា Repeater ។ Repeater ធ្វើឲ្យសញ្ញាដែលវាទទួលបានមានកម្លាំងខ្លាំងឡើងវិញហើយ Broadcast បន្តទៅមុខទៀត។ Repeater អាចជាឧបករណ៍មួយដាច់ដោយឡែកឬស្ថិតនៅក្នុង Concentrator ។ វាត្រូវបានប្រើនៅពេលដែលប្រវែងនៃខ្សែសរុបរបស់ ណេតវើកទៅតាមស្តង់ដានៃប្រភេទខ្សែដែលត្រូវប្រើ។

ឧទាហរណ៍ល្អនៃការប្រើ Repeater គួរតែប្រើនៅក្នុង LAN ដែលមានរូបរាងជា Star ជាមួយខ្សែ UTP ។ ប្រវែងកំនត់នៃ UTP គឺ១០០ម៉ែត្រ។ ការដំឡើងសម្រាប់ Workstation នីមួយៗត្រូវបានភ្ជាប់ជាមួយ UTP ទៅកាន់ Port នៃ Active Concentrator ។ Concentrator បង្កើនកំលាំងសញ្ញាលទាំងអស់ដែលបានឆ្លងកាត់វាហើយវាអនុញ្ញាតឲ្យការតភ្ជាប់នៃខ្សែអាចលើសពី១០០ម៉ែត្រ។

១-៤-៧-Bridges

TinyBridge-100

Miniature Remote Ethernet Bridge Extender



Bridges គឺជាឧបករណ៍ដែលអនុញ្ញាតឲ្យយើងអាចចែក ណេតវើកដ៏ធំមួយជាផ្នែកតូចៗជាច្រើនដែលគេហៅថា Segment ដើម្បីធ្វើឲ្យ ណេតវើកដំណើរការប្រសើរ។ បើយើងបន្ថែម Segment ថ្មីចំពោះគម្រោងនៃ ណេតវើកដែលមានស្រាប់ នោះយើងត្រូវប្រើ Bridges ដើម្បីភ្ជាប់ ណេតវើកទាំងពីរនោះ។ Bridges អាចបង្ហាញពីចរាចរណ៍ព័ត៌មានដែលស្ថិតនៅក្នុងផ្នែកទាំងសងខាងនៃ Nerworks។ ដូច្នោះវាអាចបញ្ជូន Packets ទៅចំគោលដៅច្បាស់លាស់។ Bridge ភាគច្រើនបំផុតអាចដឹងពីប្រព័ន្ធ ណេតវើកហើយវាអាចស្គាល់ Address របស់កុំព្យូទ័រនីមួយៗដែលស្ថិតនៅទាំងសងខាងរបស់វាជាស្វ័យប្រវត្តិ។ Bridge ត្រួតពិនិត្យទៅលើសារនីមួយៗហើយនៅក្នុងករណីចាំបាច់វា broadcast សារនោះទៅផ្នែកណាមួយនៃ Network។ Bridge គ្រប់គ្រងទៅលើចរាចរណ៍របស់ ណេតវើកដើម្បីធ្វើឲ្យដំណើរការរបស់ Network ដែលស្ថិតនៅផ្នែកទាំងពីររបស់វាបានដំណើរការបានល្អប្រសើរ។ គេអាចប្រើប្រាស់វាដើម្បីភ្ជាប់ ណេតវើកដែលមានខ្សែប្រភេទផ្សេងគ្នា ប្រភេទនៃរូបរាងផ្សេងគ្នា ប៉ុន្តែត្រូវបាន Software protocol ដូចគ្នា។

១-៤-៨-Router

គ្រប់ Cisco Routerមាន IOS(Internetworking Operating System)ដែលប្រើសម្រាប់គ្រប់គ្រងលើវា ។ មូលដ្ឋានគ្រឹះនៃ Cisco Router មួយចំនួនមានដូចជា

1- Interface :គឺជា Ports ដែលអនុញ្ញាតឲ្យយើងអាចភ្ជាប់ជាមួយវាបាន ។

Interfaces ផ្នែកខាងក្រោយរបស់ Cisco Router

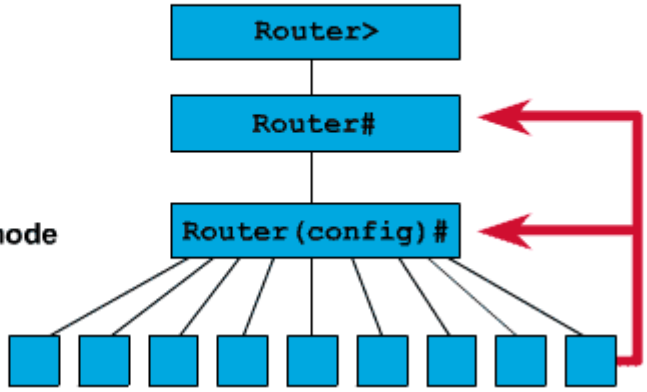
ISDN interfaces របស់ Cisco Router

Ethernet interfaces របស់ Cisco Router

2- Processor (CPU): បង្កើត Interrupt Request (IRQ)ជាមួយ Electronic Components ក្នុង Router ដើម្បីធ្វើការប្រាស្រ័យទាក់ទងជាមួយគ្នា។ Cisco Router ប្រើ Motorola RISC processor ដែលជាទូទៅការប្រើប្រាស់របស់វាមិនលើសពី 20% ។
3- IOS: គឺជាប្រព័ន្ធប្រតិបត្តិការដ៏សំខាន់ដែល Router ប្រើសម្រាប់ដំណើរការ។ វាចាប់ផ្តើមដំណើរការនៅពេលដែល Router boot ឡើង។ ជាទូទៅវាមានទំហំពី 2MB ទៅ 5MB ។
4- Rxboot image:គឺជា bootloader ដែលជា Version របស់ IOS ហើយស្ថិតនៅក្នុង ROM របស់ កុំព្យូទ័រ ។ បើគ្មាន Flash Card សម្រាប់ Load ពី IOS ទេ។ យើងអាច Configure វាឲ្យ load ចេញពី Rxboot Image ដែលផ្តល់លទ្ធភាពឲ្យយើងអាចមើលថែទាំ System និងធ្វើឲ្យ Interfaces ផ្សេងៗទៀតអាចដំណើរការឬមិនអាចដំណើរការបាន ។
5- NVRAM (Non-volatile RAM):គឺជា Memory ពិសេសដែលវារក្សាទុក Configuration របស់វា។ នៅពេលយើង Configure វាហើយ Save វាត្រូវបានរក្សាទុកនៅទីនោះ (NV RAM)។
6- ROM (Read-Only Memory): គឺវារក្សាទុកនូវ Bootrap និង POST ។
7- Flash Memory: គឺជាEEPROM (Electronic Erasable Programmable Read Only Memory)
8- Configuration Register: សម្រាប់កំណត់ថាតើ Router ត្រូវ boot IOS image ពីFlash (tftp Server) ឬគ្រាន់តែ Load ចេញពី Rxboot image ។

Overview of Router Modes

- User Exec mode
- Privileged Exec mode
- Global configuration mode
- Specific Configuration modes



Configuration Mode	Prompt
Interface	`Router(config-if)#`
Subinterface	`Router(config-subif)#`
Controller	`Router(config-controller)#`
Map-list	`Router(config-map-list)#`
Map-class	`Router(config-map-class)#`
Line	`Router(config-line)#`
Router	`Router(config-router)#`
IPX-router	`Router(config-ipx-router)#`
Route-map	`Router(config-route-map)#`

ប្រភេទនៃ Cisco Routers មានដូចខាងក្រោម

Cisco 700 series

-Cisco 800 series

- Cisco 1600 series

Cisco 2600 series

Cisco 7200 Series

តួនាទីរបស់ Router

សម្រាប់បកប្រែព័ត៌មានពី ណេតវើកមួយទៅកាន់ nerwork មួយផ្សេងទៀត វាហាក់បីដូចជា Bridge ដ៏វែឆ្ងាតមួយបំផុត។ Router អាចជ្រើសរើសផ្លូវបញ្ជូនដ៏ប្រសើរបំផុតដើម្បីបញ្ជូន Packets ដោយពឹងផ្អែកទៅលើ Address របស់អ្នកទទួលនិងអ្នកបញ្ជូនតាមផ្លូវណាដែលខ្លីជាងគេបំផុត។ ចំណែកឯ Bridge វិញស្គាល់ Address របស់កុំព្យូទ័រដែលស្ថិតនៅក្នុងផ្នែកនីមួយៗនៃ ណេតវើក(MAC)។

Router ស្គាល់ Address របស់កុំព្យូទ័រ Bridge និង Address របស់ Router ផ្សេងៗទៀតរបស់ប្រព័ន្ធ Network។ Router មិនគ្រាន់តែដឹងពីប្រព័ន្ធ ណេតវើកទាំងមូលនោះទេ ប៉ុន្តែវាអាចដឹងពីផ្នែកណាមួយនៃ ណេតវើកដែលវែល់ជាងគេបំផុត។

បើសិនជា LAN នៅសាលាភ្ជាប់ទៅនិង Intenrnet នោះគេត្រូវទិញ Router។ នៅក្នុងករណីនេះ Router ដើរតួនាទីជាអ្នកបកប្រែព័ត៌មានដែលស្ថិតនៅលើ LAN ជាមួយប្រព័ន្ធ Internet។ Router អាចជ្រើសរើសផ្លូវដែលល្អប្រសើរបំផុតដើម្បីបញ្ជូនទិន្នន័យតាមប្រព័ន្ធ Internet។ Routerអាចធ្វើការបានដូចខាងក្រោម:

- វាអាចដឹកនាំចារចរណ៍របស់សញ្ញាលដោយផ្ទាល់ហើយមានប្រសិទ្ធភាពខ្ពស់
- វាអាចបញ្ជូនសាររវាង Protocol ពីរផ្សេងគ្នា
- វាអាចបញ្ជូនសាររវាងរូបរាងជា bus star និង Star-wired ring
- វាអាចបញ្ជូនសារតាម Fiber Optic Coaxial និង UTP

១-៥-តើ ណេតវើកCabling ជាអ្វី?

Cable គឺជាខ្សែដែលផ្តល់ឲ្យព័ត៌មានផ្លាស់ប្តូរពីឧបករណ៍មួយទៅឧបករណ៍មួយទៀត។ ខ្សែមាន៤ប្រភេទដែលអាចប្រើជាមួយ LAN បាន។ ក្នុងករណីមួយចំនួន ណេតវើកប្រើខ្សែមួយប្រភេទហើយ ណេតវើកប្រភេទផ្សេងទៀតប្រើខ្សែប្រភេទទៀងទៀតដែរ។ ប្រភេទនៃខ្សែដែលត្រូវបានជ្រើសរើសសម្រាប់ ណេតវើកទាក់ទងជាមួយនិងរូបរាងនៃ ណេតវើកនិងទំហំរបស់វា។

ការយល់ដឹងពីលក្ខណៈខុសគ្នានៃខ្សែនិងរបៀបដែលវាទាក់ទងទៅនិងទិដ្ឋភាពផ្សេងៗនៃ ណេតវើកមួយចាំបាច់សម្រាប់អភិវឌ្ឍជោគជ័យទៅលើ ណេតវើកទាំងមូល។ នៅផ្នែកខាងក្រោមនេះពិភាក្សាទៅលើប្រភេទនៃខ្សែដែលត្រូវប្រើក្នុង ណេតវើកនឹងអ្វីៗដែលទាក់ទងជាមួយនឹងប្រធានបទ។

- Unshielded Twisted Pair (UTP) Cable
- Shield Twist Pair (STP) Cable
- Coaxial Cable
- Fiber Optic Cable
- Wireless LANs
- Cable installation Guides

១-៥-១-Unshielded Twisted Pair (UTP) Cable

ខ្សែ Twisted Pair មានពីរប្រភេទគឺ **Shielded** និង **Unshield**ded ។ **Unshielded Twisted Pair (UTP)** ត្រូវបានគេប្រើច្រើននិងជាទូទៅវាត្រូវបានគេប្រើល្អបំផុតសម្រាប់ ណេតវើកនៅតាមសាលារៀន។

ខ្សែ **Cat7**

គុណភាពនៃខ្សែ UTP អាចខុសគ្នាពីខ្សែសម្រាប់ប្រព័ន្ធទូរសព្ទ៍ពីព្រោះវាមានល្បឿនលឿន។ ខ្សែនេះមាន៤គូរដែលស្ថិតនៅក្នុងសំបករបស់វា។ គូនីមួយៗត្រូវបានវិញបញ្ចូលគ្នាតាមស្តង់ដាដើម្បីជួយបំបាត់នៃការរំខានពីគូដែលជាប់គ្នានិងឧបករណ៍អេឡិចត្រូនិកផ្សេងៗទៀត។

EIA/TIA (Electronic Industry Association/Telecommunication Industry Association) បានបង្កើតស្តង់ដានៃ UTPនឹងតម្លៃ៧ប្រភេទនៃខ្សែ។

ប្រភេទ	ការប្រើប្រាស់
Category 1	Voice Only (Telephone Wire)
Category 2	Data to 4 Mbps (LocalTalk)
Category 3	Data to 10 Mbps (Ethernet)
Category 4	Data to 20 Mbps (16 Mbps Token Ring)
Category 5	Data to 100 Mbps (Fast Ethernet)
Category 6	Data to 1000Mbps(Gigabit Ethernet)
Category 7	Data to 100Gbps

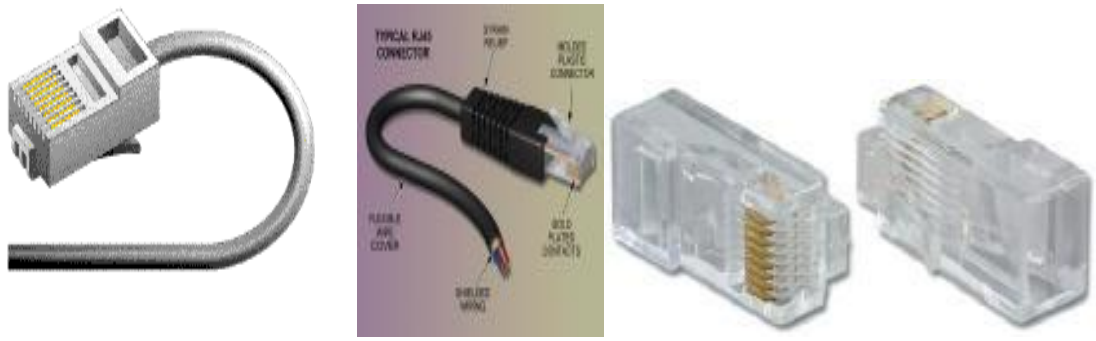
រូបភាព៥-បង្ហាញពីការសង្ខេបនៃ Categories of Unshielded Twisted Pair

ប្រសិនបើអ្នករៀបចំ 10 Mbps Ethernet ណោតវើកនិងគិតដល់ការសន្សំក្នុងការទិញខ្សែ អ្នកគួរទិញខ្សែប្រភេទទី៣ជំនួសឲ្យប្រភេទទី៥វិញ។ ត្រូវចាំថាខ្សែប្រភេទទី៥ផ្តល់នូវលទ្ធភាពជាច្រើនដូចជា "នៅពេលដែលមានការកើនឡើងនៃចំនួនឧបករណ៍" ជាដើម។ ខ្សែប្រភេទទី៣និងទី៥ មានប្រវែង Segment វែងបំផុតគឺ១០០ម៉ែត្រ។

- 10BaseT សំដៅទៅលើទំហំនៃខ្សែ UTP (cat 3,4,5) សម្រាប់ទទួល Ethernet signal
- ប្រភេទទី៧គឺជាប្រភេទថ្មីសម្រាប់ Gigabit Ethernet ។

Unshielded Twisted pair Connector

ស្តង់ដានៃ Connector សម្រាប់ UTP គឺ RJ-5 ។ វាគឺជាក្បាលប្រភេទជាប្លាស្ទីកដែលមើលទៅដូចទៅនិងក្បាលសម្រាប់ភ្ជាប់នៃទូរសព្ទដែរ។ Slot មួយអនុញ្ញាតឲ្យស៊កបាន RJ-45 តែមួយគត់។ RJ-5 តំណាងឲ្យ Registered Jack ។



រូបភាព៦-បង្ហាញពី RJ-45 connectors

Shielded Twisted Pair (STP) Cable[i]

Cable	Length ft.	Max. Resolution
Cat5e Solid UTP	100	2048 x 1536 at 60Hz
	150	2048 x 1536 at 60Hz
	200	2048 x 1536 at 60Hz
Cat5e Stranded UTP/STP	100	2048 x 1536 at 60Hz
	200	1280 x 1024 at 60Hz
Cat6 Solid UTP	100	1920 x 1440 at 60Hz
	150	1600 x 1200 at 60Hz
550MHz Cat6 Solid UTP	100	2048 x 1536 at 60Hz
	150	1920 x 1200 at 60Hz
Cat6 Solid STP	100	2048 x 1536 at 60Hz
	150	1920 x 1200 at 60Hz
600MHz Cat7 Solid STP	150	2048 x 1536 at 60Hz
	200	2048 x 1536 at 60Hz
600MHz Cat7 Stranded STP	100	2048 x 1536 at 60Hz

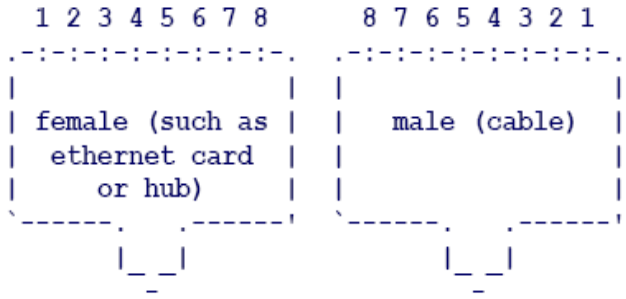
គុណវិបត្តិនៃខ្សែ UTP គឺថាវាអាចងាយប៉ះពាល់ជាមួយប្រេកង់នៃវិទ្យុនិងចរន្តអគ្គសនី។ Shielded twisted pair (STP)មានលក្ខណៈសមរម្យសម្រាប់មជ្ឈដ្ឋានដែលមានការរំខាន។ Shielded twisted pair ត្រូវបានប្រើសម្រាប់ Token Ring topology ។

RJ-45 colors

RJ-45 cable និង plug មាន 8 pin/conductors ។ Pins ទាំងនោះត្រូវបានប្រើជាគូ ។ វាមានសម្មតិកម្មអំពីរបៀបរៀបគូនៃ Pin ប៉ុន្តែ 100 baseT ត្រូវបានគេប្រើច្រើនជា EIA 568B ។ វាមានសម្មតិកម្មកូដបីពណ៌សម្រាប់អង្គធាតុចំលងនីមួយៗជាមួយ RJ-45 cable ។

ក្នុងការតភ្ជាប់ខ្សែ UTP ជា Straight Cable គេអាចភ្ជាប់តាមពីររបៀបគឺស្តង់ដា A និងស្តង់ដា B ។
ស្តង់ដា EIA 568B

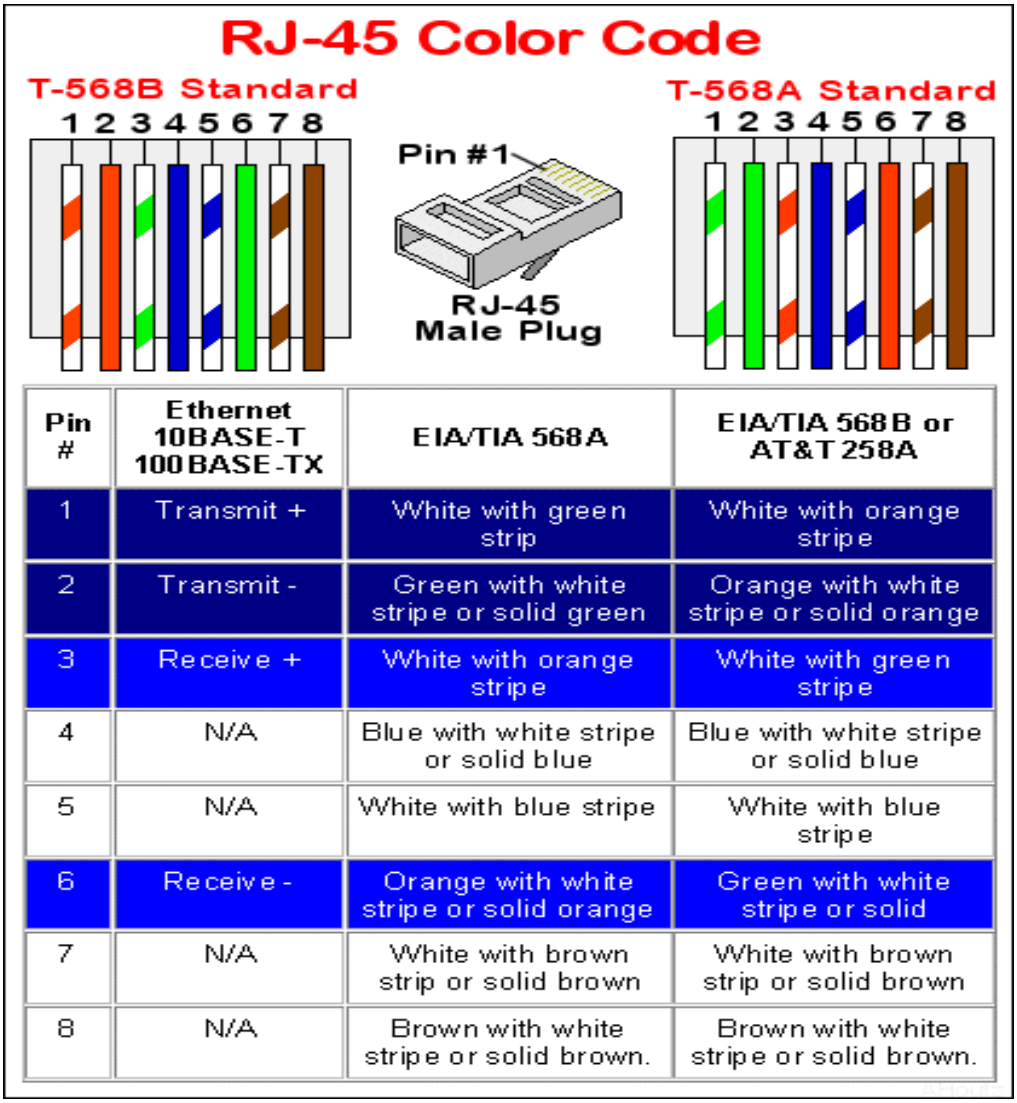
- Pin លេខ១គេដោតពណ៌សឆ្នូតទឹកក្រូចនឹង Pin លេខ២ គេដោតពណ៌ទឹកក្រូចដែលគេហៅថាគូទី២ ។
- Pin លេខ៣ គេដោតពណ៌សឆ្នូតបៃតងនិង Pin លេខ៦ត្រូវបានគេដោតពណ៌បៃតងដែលគូរនេះគេហៅថាគូទី៣
- Pin លេខ៤ត្រូវបានគេដោតពណ៌ខៀវនិង Pin លេខ៥ត្រូវបានគេដោតពណ៌ឆ្នូតខៀវដែលគេហៅថាគូទី១
- Pin លេខ៧ត្រូវបានគេដោតពណ៌ឆ្នូតត្នោតនិង Pin លេខ៨ ត្រូវបានគេដោតពណ៌ត្នោតដែលគេហៅថាគូទី៤



រូបភាព- Connector

- TIA/EIA 568 B (សមមូលនិង DTE)
  - Leg 1: ពណ៌សឆ្នូតទឹកក្រូច
  - Leg2: ទឹកក្រូច
  - Leg3: ពណ៌សឆ្នូតបៃតង
  - Leg 4: ពណ៌បៃតង
  - Leg5: ពណ៌សឆ្នូតខៀវ
  - Leg 6: ពណ៌ខៀវ
  - Leg 7: ពណ៌សឆ្នូតត្នោត
  - Leg 8: ពណ៌ត្នោត
- TIA/EIA 568 A (សមមូលនិង DCE)
  - Leg 1: ពណ៌សឆ្នូតបៃតង
  - Leg2: ពណ៌បៃតង
  - Leg3: ពណ៌សឆ្នូតទឹកក្រូច
  - Leg 4: ពណ៌ខៀវ
  - Leg5: ពណ៌សឆ្នូតខៀវ
  - Leg 6: ទឹកក្រូច

- Leg 7: ពណ៌សឆ្នូតខ្មៅ
- Leg 8: ពណ៌ខ្មៅ



Pin #	Ethernet 10BASE-T 100BASE-TX	EIA/TIA 568A	EIA/TIA 568B or AT&T 258A
1	Transmit +	White with green strip	White with orange stripe
2	Transmit -	Green with white stripe or solid green	Orange with white stripe or solid orange
3	Receive +	White with orange stripe	White with green stripe
4	N/A	Blue with white stripe or solid blue	Blue with white stripe or solid blue
5	N/A	White with blue stripe	White with blue stripe
6	Receive -	Orange with white stripe or solid orange	Green with white stripe or solid
7	N/A	White with brown strip or solid brown	White with brown strip or solid brown
8	N/A	Brown with white stripe or solid brown.	Brown with white stripe or solid brown.

ឧបករណ៍សម្រាប់តេស្តខ្សែ

ឧបករណ៍សម្រាប់តេស្តខ្សែ CAT 6

១-៦-Cisco 16xx/26xx/36xx Routers

Cisco Router មាន RI Console Port ហើយវាដំណើរការនូវ IOS (Internet Working Operating System) គ្របប្រភេទទាំងអស់។ នៅក្នុងមេរៀននេះយើងនិងសិក្សាទៅលើស៊េរីនៃ 1600s និង 2600s តែប៉ុណ្ណោះ។ Station Card និង Router ស៊េរី 2600 មាន RJ-45 ប៉ុន្តែវាទាំងពីរមិនត្រូវបានប្រើប្រាស់ជាមួយគ្នានោះទេ។ ដូច្នោះវាត្រូវតែមាន RJ-45 ទាំងសងខាងហើយយើងត្រូវតែដោតខ្សែនេះឲ្យបានត្រឹមត្រូវ។ តាមធម្មតាគេភ្ជាប់ Router ទៅនិង Patch Panel និង Stallion card ដោយប្រើខ្សែពិសេស។

Cisco RJ-45 Pin	Colour	Cisco Signal	Stallion RJ-45 Pin	Stallion Signal
1	White/Green	RTS	N/C	
2	Green	DTR	N/C	
3	White/Orange	TxD	5	RxD
4	Blue	Gnd	3	Gnd
5	White/Blue	Gnd	6	Gnd
6	Orange	RxD	4	TxD

Cisco RJ-45 Pin	Colour	Cisco Signal	Stallion RJ-45 Pin	Stallion Signal
7	White/Brown	DSR	N/C	
8	Brown	CTS	N/C	

រូបភាព-៧-Cisco RJ-45

១-៧-Cisco Catalyst Switches

ការចាប់គូ Pin នៅលើ Cisco Console Port របស់ Switch Catalyst ខុសគ្នាពីការចាប់គូ Pin នៅលើ Cisco 26xx ។ វិធីដែលគេចាបអារម្មណ៍គឺអាស្រ័យទៅលើ Operating software ។

Specification	Cable Type	Maximum length
10BaseT	Unshielded Twisted Pair	100 meters
10Base2	Thin Coaxial	185 meters
10Base5	Thick Coaxial	500 meters
10BaseF	Fiber Optic	2000 meters
100BaseT	Unshielded Twisted Pair	100 meters
100BaseTX	Unshielded Twisted Pair	220 meters

រូបភាព-៨-បង្ហាញពី Ethernet

សំណួរត្រួតពិនិត្យ

១-តើ Networking Hardware មានអ្វីខ្លះ?

២-ចូរបញ្ជាក់ពីតួនាទីនីមួយៗរបស់ Networking Hardware?

៣-តើ Switch និង router ខុសគ្នាដូចម្តេច?

៤-តើ Router មួយមានសមាសធាតុផ្សំអ្វីខ្លះ?

៥-តើ Router មានតួនាទីអ្វីខ្លះក្នុងប្រព័ន្ធ Network?

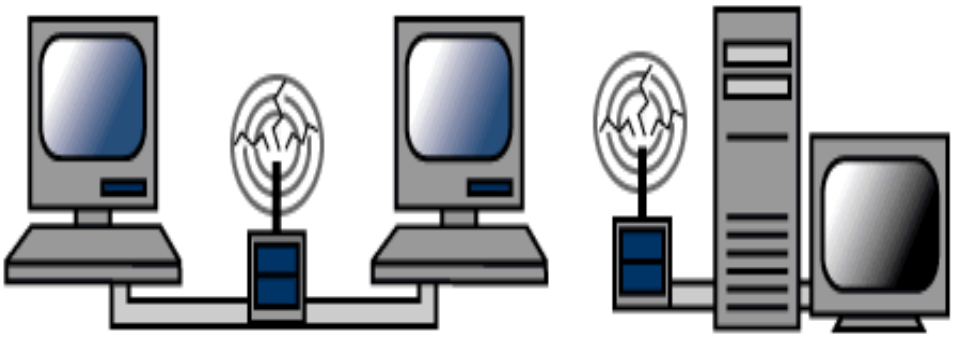
៦-តើខ្សែ UTP មានប៉ុន្មាន Category?ហើយ Category នីមួយៗខុសគ្នាដូចម្តេច?

៧-តើ Server (Hardware)មួយទាមទារឲ្យមានសមាសធាតុផ្សំអ្វីខ្លះ?

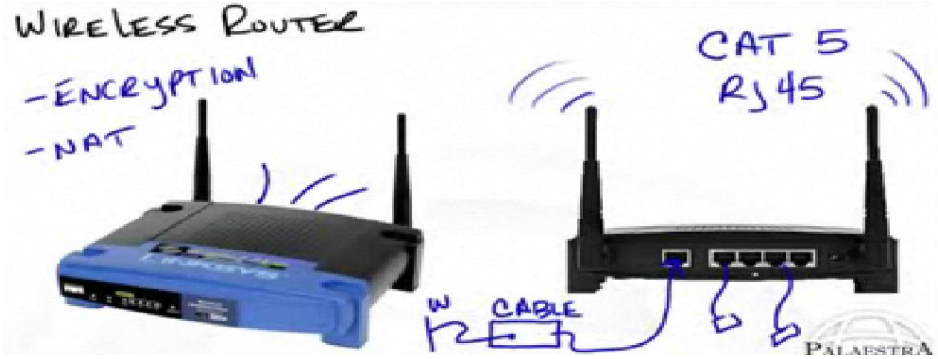
៨-តើ Pin ទាំង៨របស់ RJ-45 connector មានតួនាទីអ្វីខ្លះ?

១-៨-Wireless LANs

ដើម្បីភ្ជាប់ Wireless LANs បានគេត្រូវមាន Wireless ណេតវើកCard និង Access Point (AP)។



១-៨-១-ប្រភេទនៃ Access Point (AP)



Wireless Router អាចដើរតួនាទីជា DHCP, NAT, Firewall ។

គ្រប់ ណេតវើកទាំងអស់មិនមែនតភ្ជាប់គ្នាដោយប្រើខ្សែទាំងអស់នោះទេ។ មាន ណេតវើកប្រភេទខ្លះត្រូវបានគេតភ្ជាប់ដោយមិនប្រើខ្សែដែលហៅថា Wireless LAN គឺប្រើរលកសញ្ញាលរបស់វិទ្យុដែលមានប្រេកង់កម្រិតខ្ពស់ដូចជាបាច់ពន្លឺ Infrared ឬកាំរស្មី Laser ដើម្បីបញ្ជូនរវាង Workstation និង Fileserver ដែលស្ថិតនៅលើ Wireless ណេតវើកដោយប្រើនូវ Transceiver ឬជាអង់តែនដែលប្រើសម្រាប់បញ្ជូននិងទទួលទិន្នន័យ។

ព័ត៌មានត្រូវបានបញ្ជូនឆ្លងកាត់តាម Transceiver នៅពេលដែលគេតភ្ជាប់វាតាមលក្ខណៈជារូបរាងនៃ ណេតវើកដែលមានចម្ងាយឆ្ងាយ ។ ការប្រាស្រ័យទាក់ទងគ្មានខ្សែអាចប្រើប្រាស់ដូចជា ទូរសព្ទ ម៉ាយក្រូវ៉ែវ ផ្កាយរណប។ ណេតវើកគេប្រើ

សម្រាប់ Laptop កុំព្យូទ័រឬកុំព្យូទ័រពីចម្ងាយដែលភ្ជាប់ទៅនិង LAN ដែលគ្មានខ្សែដែលផ្តល់ឲ្យនូវគុណសម្បត្តិក្នុងការតភ្ជាប់ប្រព័ន្ធ ណេតវើកដែលស្ថិតនៅក្នុងអាគារចាស់ៗដែលយើងមិនអាចអូសខ្សែបានបាន។ ការប្រាស្រ័យទាក់ទងតាម Infrare ដែលគេនិយមប្រើក្នុងសាលារៀនមានពីរប្រភេទគឺបាច់ពន្លឺខ្សែត្រង់(lightsight)និងបាច់ពន្លឺបញ្ជូនទៅគ្រប់ទីកន្លែង(scattered broadcast light)។

បើសិនជានរណាម្នាក់ដើរកាត់បាច់ពន្លឺខ្សែត្រង់នោះក្នុងពេលវាកំពុងធ្វើការបញ្ជូនទិន្នន័យ នោះព័ត៌មានត្រូវបញ្ជូនឡើងវិញសាជាថ្មីដែលឧបសគ្គបែបនេះធ្វើឲ្យប្រព័ន្ធឥតខ្សែ(Wireless) ដើរយឺត។

ការប្រាស្រ័យទាក់ទងរបស់ Infrare ជារបាយគឺវាធ្វើឲ្យការបញ្ជូនទិន្នន័យតាមទម្រង់ជា Broadcast មានន័យថាសញ្ញាលត្រូវបានបញ្ជូនតាមទិសដៅជាច្រើនហើយអាចបត់បែនតាមជញ្ជាំងឬពិដានរហូតដល់ឧបករណ៍ Transceiver ។ ការប្រាស្រ័យទាក់ទងក្នុងប្រព័ន្ធ ណេតវើក ដែលប្រើ Laser គឺជាបាចពន្លឺនៃ infrared ដែលជាបន្ទាត់ខ្សែត្រង់។

Wireless LAN មានគុណវិបត្តិដូចខាងក្រោម:

- មានសុវត្ថិភាពទាប
- ងាយឆ្លងពីប្រភពពន្លឺនឹងឧបករណ៍អេឡិចត្រូនិក
- មានល្បឿនយឺតជាង LAN ដែលប្រើខ្សែ

Wireless មាន Standard IEEE 802.11

- IEEE 802.11.b–2.4 GH_Z -11 Mbps មានចម្ងាយ 45mពី access point
- IEEE 802.11.g-2.4GH_z-Hybrid/54Mbps
- IEEE 802.11.N:ចាប់ពី 54Mbps ឡើងទៅ
- IEEE 802.11.a:54Mbps

១-៨-២-ការរៀបចំដើម្បី Setup Wireless ណេតវើក

Wireless Router ឬ Wireless Access Point

Wireless Router ឬ Wireless Access Point គឺជាឧបករណ៍ដ៏សំខាន់មួយនៅពេលដែលយើងត្រូវការ Setting Up ចំពោះ Wireless network:

- យើងប្រើ Wireless Router នៅពេលដែលយើងបង្កើងត ណេតវើកជាលើកដំបូងនៅតាមផ្ទះឬការិយាល័យ ។ បើយើងមាន Ethernet ណេតវើកនៅតាមផ្ទះហើយ យើងគ្រាន់តែទិញនូវ Wireless Access Point និងគ្រាន់តែភ្ជាប់វាជាមួយណេតវើកនោះជាការស្រេច។

Wireless Adapter for កុំព្យូទ័រ Users

- យើងត្រូវមាននូវឧបករណ៍ Wireless adater ដែលត្រូវបានដំឡើងរួចជាស្រេចហើយនៅលើកុំព្យូទ័រ របស់អ្នកក្នុងគោលបំណងចូលរួមក្នុងប្រព័ន្ធ ណេតវើកដែលគ្មានខ្សែ។

អ្នកប្រើប្រាស់កុំព្យូទ័រឬ Notebook ហើយអ្នកត្រូវការ Share នូវ Wireless adapter ជាមួយកុំព្យូទ័រផ្សេង នោះអ្នកត្រូវការប្រើនូវ USB wireless Adapter ។ អ្នកគ្រាន់តែដោត USB Wireless Adapter នៅលើកុំព្យូទ័រដែល អ្នកត្រូវការឲ្យមាន Wireless Acces ។

១-៨-៣-វិធីរហ័សដែលឃើញ Wireless ណេតវើកProfile នៅក្នុង Windows Vista

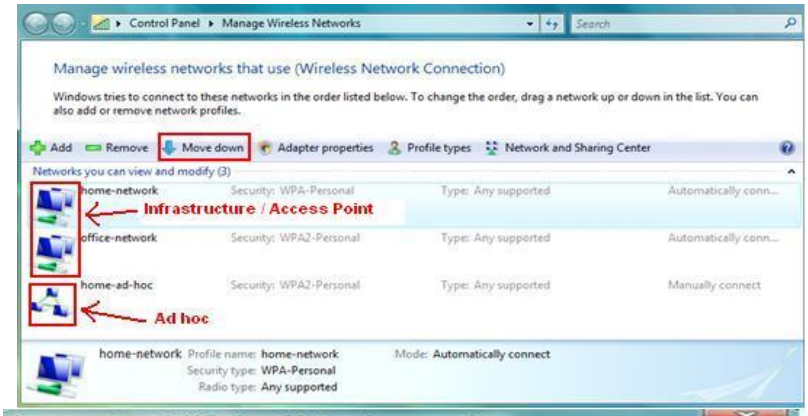
1) ចុចលើ **Start** ហើយចុចដោយប្រើប៊ូតុង Mouse ខាងស្តាំលើ **ណេតវើក**ហើយបន្ទាប់មកចុចលើ **Properties**.

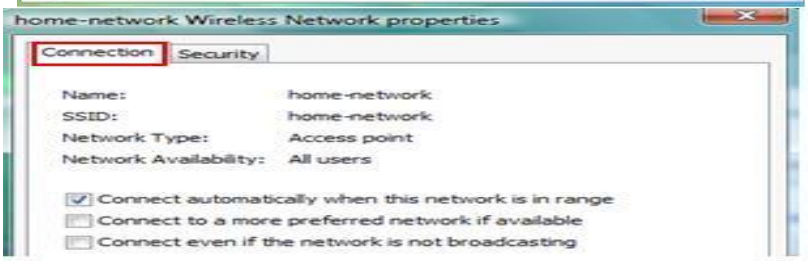
2) ណេតវើកand Sharing Center window will appear, then click **Manage wireless networks** on the left panel.

3) Manage Wireless Networks window និងបង្ហាញឡើងហើយអ្នកនិងឃើញ wireless ណេតវើក connection មាន។ ចុចពីរដងស្ទួនគ្នាលើ profile ហើយបន្ទាប់មកអ្នកអាចឃើញឬអាចប្តូរ setting របស់ វា

ចំណាំ:កុំព្យូទ័រ icon ពីរធំៗបង្ហាញថាវាគឺជា infrastructure ឬ access point network មានន័យថាអ្នកនិងភ្ជាប់ ណេតវើកតាមរយៈ access point ឬ wireless router។ រូបភាពកុំព្យូទ័រតូចៗចំនួន៣គឺជា ad hoc wireless ណេ តវើកមានន័យថា កុំព្យូទ័រs និងភ្ជាប់គ្នាទៅវិញទៅមកដោយផ្ទាល់។

ចំណាំ:ណេតវើកprofile ដែលស្ថិតនៅទីតាំងខ្ពស់ជាងនិងមានអាទិភាពជាង។ ឧទាហរណ៍ home-ណេតវើកនិង ត្រូវបានភ្ជាប់ជំនួសឲ្យ office-ណេតវើកបើ networks ទាំងពីរត្រូវបានភ្ជាប់គ្នាចំពោះអទិភាពជាងនៅលើ home-ណេតវើកprofile។ អ្នកអាច **Move up/Move down** Icon ដើម្បីកំណត់អាទិភាព។



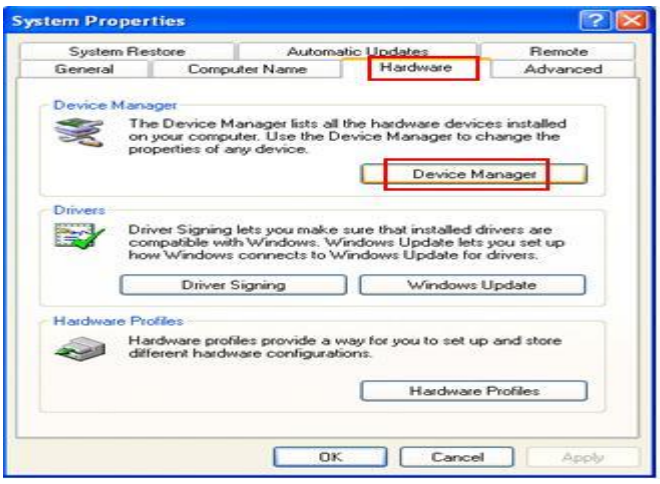


4) វិធីផ្សេងទៀតដែលអ្នកអាចប្តូរឬមើលពី connection profileលើ window(**Start-> Connect To**) ដោយចុចលើប៊ូតុងខាងស្តាំលើ wireless connection ហើយចុចលើ **Properties**។ ទោះបីជាយ៉ាងណាក៏ដោយអ្នកមិនអាចមើល profile នៅលើ wireless connection ថ្មីបាននោះទេ ។

១-៨-៤វិធីដំលឿនបំផុតគឺពិនិត្យមើល Wireless Card Driver Status ក្នុង Windows XP ដោយ

1)Go to **Start**, right click **My កុំព្យូទ័រ** and then click on **Properties**.

2) System Properties window និងបង្ហាញឡើង។ ចូលមក **Hardware** tab ហើយចុចលើ **Device Manager**.

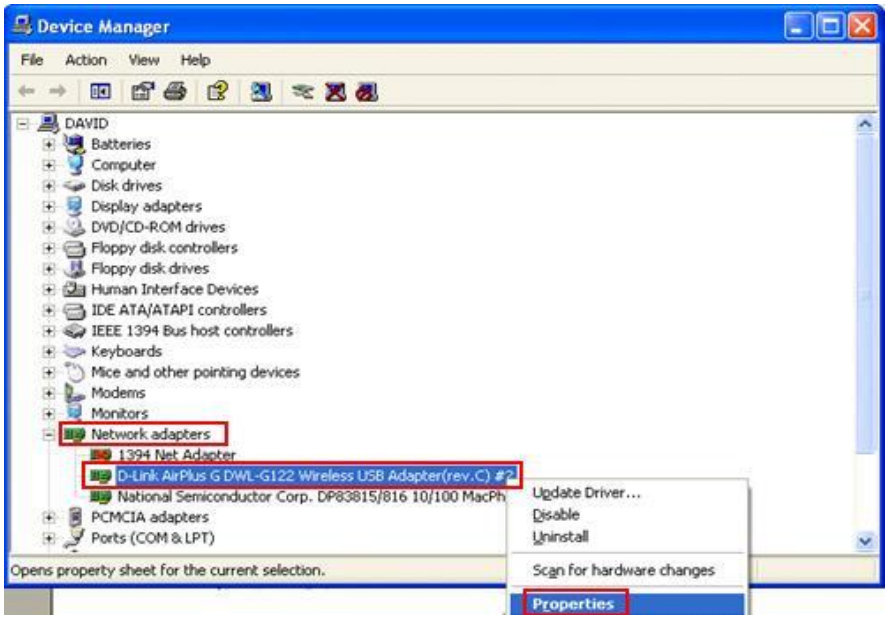


3) Device Manager window និងបង្ហាញឲ្យឃើញហើយអ្នកអាចពិនិត្យមើលពីលក្ខណៈរបស់ hardware drivers ទាំងអស់នៅទីនេះ។ ចុចទាញវាចុះក្រោមរកមើល **Network adapters** ហើយបន្ទាប់មកចុចលើប៊ូតុងខាងស្តាំលើ wireless adapter ដែលអ្នកត្រូវការពិនិត្យមើលនិងចុចលើ **Properties**។

ឧទាហរណ៍

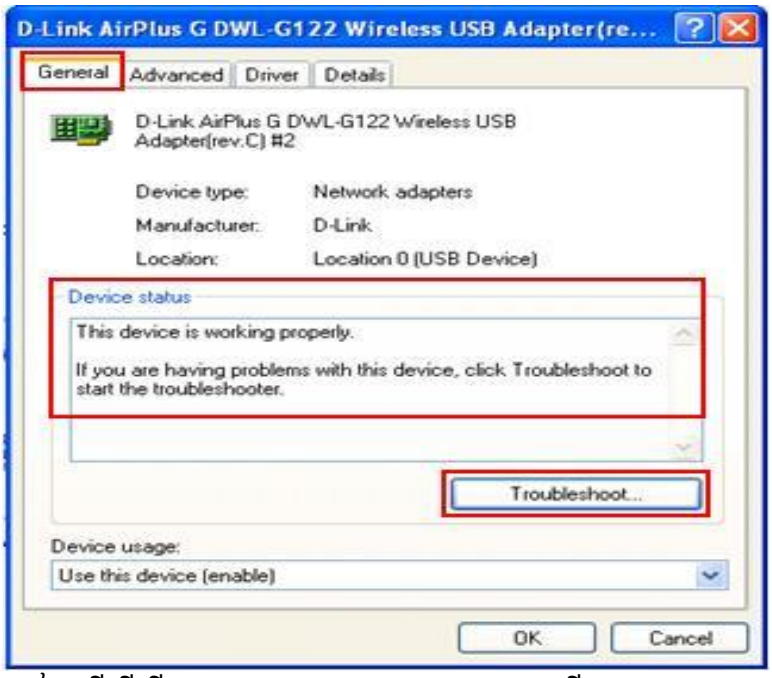
D-Link DWL-G122 USB wireless card's driver status.

ចំណាំ:បើអ្នកមិនអាចរកឃើញ wireless card នៅទីនេះទេ ចូរព្យាយាមមើលវាក្នុង **Other Devices** folder បើឃើញមានន័យថាអ្នកមិនបាន Install driver សម្រាប់ hardware នេះ ។

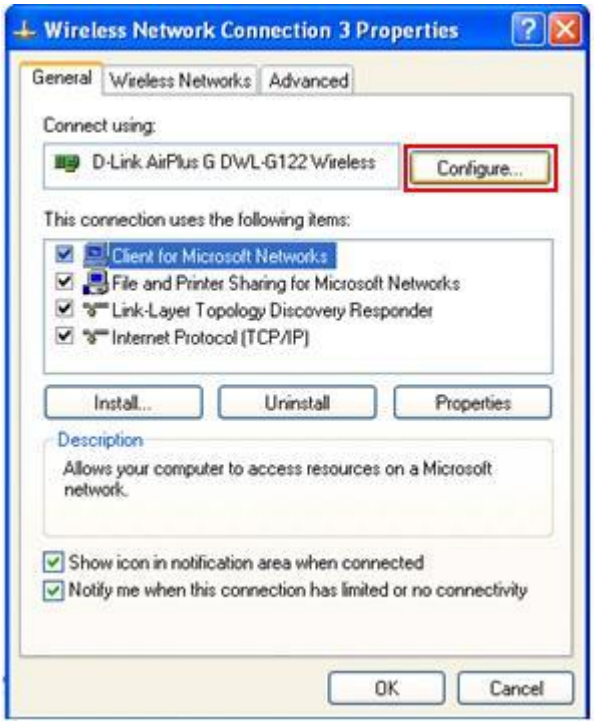


4) wireless card properties window និងបង្ហាញឡើងហើយអ្នកនិងឃើញលក្ខណៈនៃ Device បើ device មិនអាចដំណើរការបានល្អនោះទេ អ្នកអាចចុចលើ**Troubleshoot...**ដើម្បីដោះស្រាយបញ្ហាដែលបានកើតមានឡើងដែលមានការណែនាំភ្ជាប់មកជាមួយស្រាប់។

ចំណាំ:**Troubleshoot...** និងផ្តល់ដោយ Windows XP ប៉ុន្តែវាជាប្រភពនៃការសិក្សាបន្ថែមចំពោះ wireless card driver។ អ្នកអាចចចុចលើ **Advanced** tab ដើម្បីបង្កើត advanced hardware configuration,**Driver** tabដើម្បី update/roll back/uninstall driver។



ចំណាំ: គេមានវិធីផ្សេងទៀតដើម្បីមើល device status នោះដោយចុចលើ **Configure** button នៅក្នុង wireless **card** property ។



១-៨-**៥**-វិធីដំឆាប់ហ៍សដើម្បីពិនិត្យមើល ណេតវើកឬ Wireless Adapter Driver Status ក្នុង Windows Vista

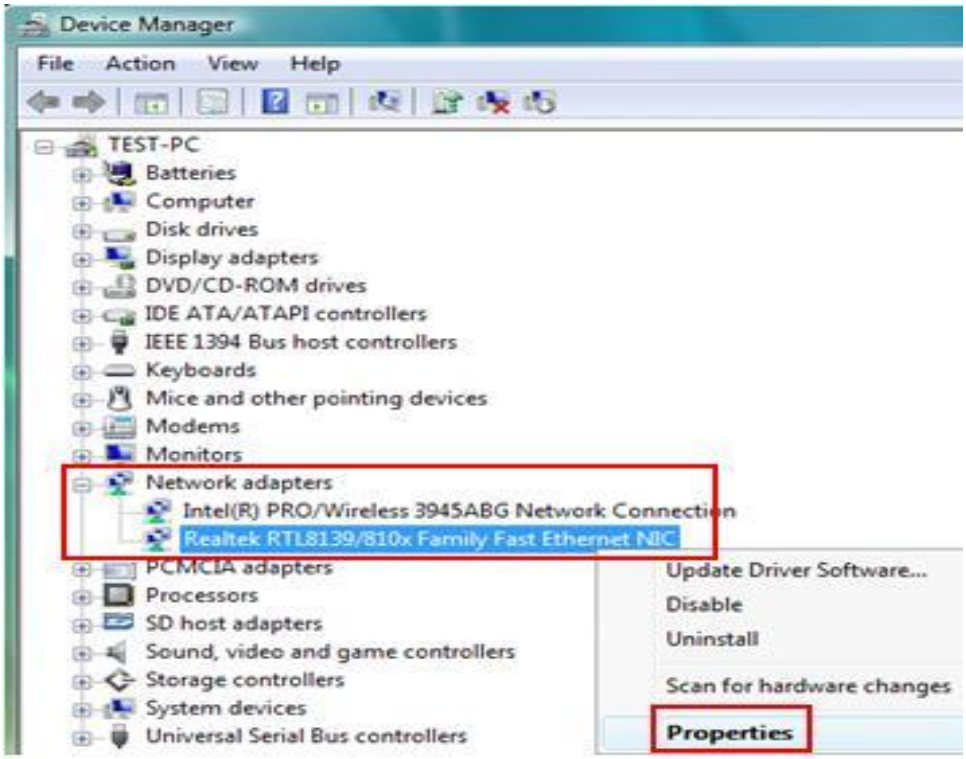
1) ចុចលើ **Start** ហើយបន្ទាប់មកចុចលើ **Control Panel**.

2) ចុចលើ **Hardware and Sound** ក្នុង Control Panel window.

ចំណាំ: បើ Control Panel ជា Classic View អ្នកគ្រាន់ចុចពីរដងលើ Device Manager.

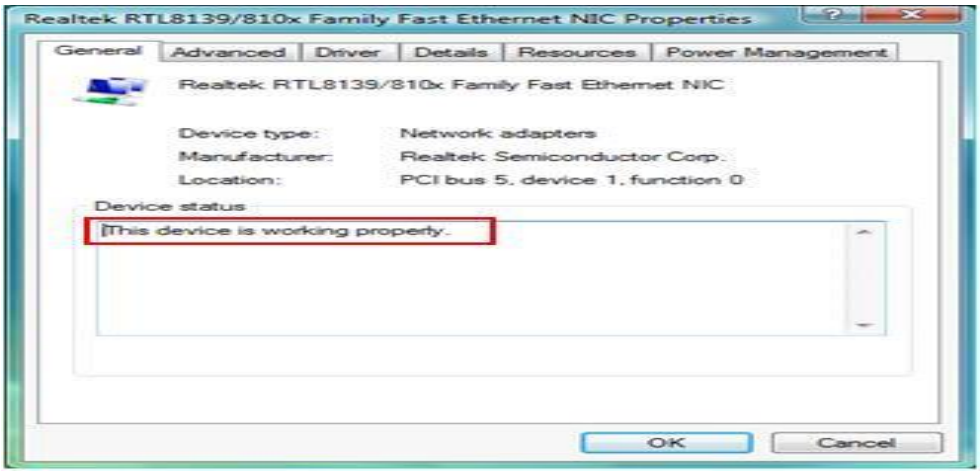
3) នៅក្នុង Hardware and Sound window ចុចលើ **Device Manager** ដើម្បីបើកវា

4) Device Manager និងបង្ហាញឡើងហើយបន្ទាប់មកពង្រីក **ណេតវើកadapters** ហើយចុចដោយប្រើប៊ូតុង Mouse ខាងស្តាំលើ ណេតវើកឬ wireless adapter ដែលអ្នកត្រូវការត្រួតពិនិត្យហើយចុងក្រោយបង្អស់ចុចលើ **Properties** ។



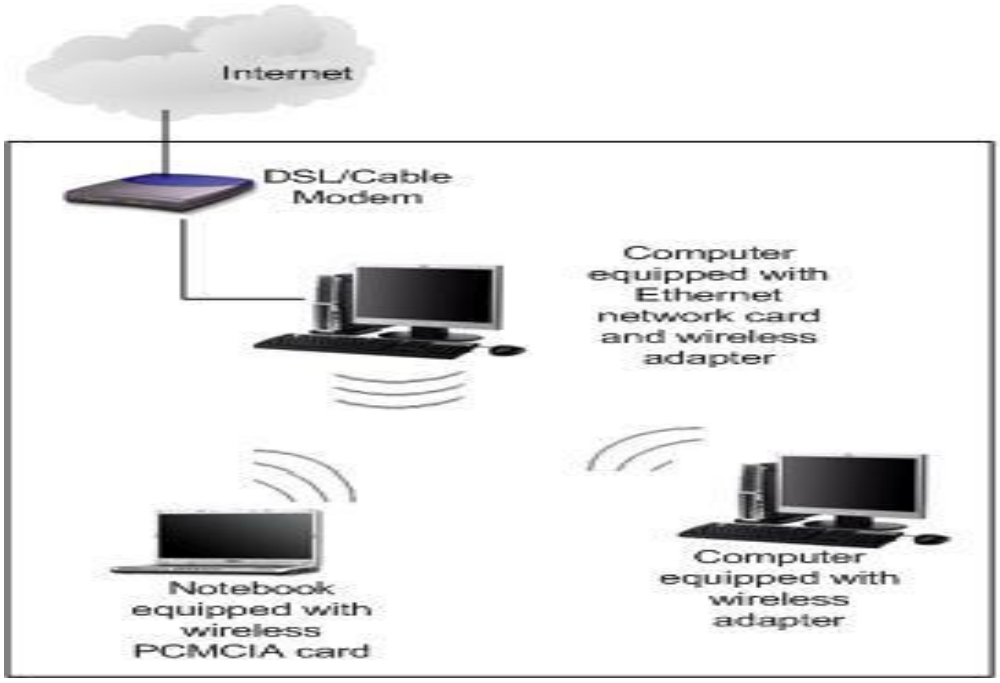
5) ណេតវើកឬwireless adapter properties window និងបង្ហាញឲ្យឃើញថា driver កំពុងដំណើរការបានល្អបើវាបង្ហាញថា**This device is working properly**នៅក្រោម General tab អ្នកក៏អាច configure ណេតវើកadapter's driver parameters ស្ថិតក្នុង Advanced tab ។

ចំណាំ:បើ device របស់អ្នកមិនដំណើរការបានល្អនោះទេ ចុចលើ Driver tab ដើម្បីពិនិត្យមើលឲ្យបានក្បោះក្បាយ update driver, disable driver ឬ uninstall driver ។



១-៨-៦-របៀប Set Up Ad Hoc Wireless ណេតវើកក្នុង Windows Vista

តាមធម្មតាអ្នកមានប្រភេទ ណេតវើកជាបណ្តោះអាសន្នដើម្បី share documents ក្នុងការប្រជុំ ការលេង game ជាច្រើននាក់ឬ share Internet connection ជាមួយមិត្តភក្តិ ។ វាមានគុណវិបត្តិមួយរបស់ ណេតវើកនេះគឺ wireless អាចភ្ជាប់បានក្នុងរយៈចម្ងាយខ្លី ។ ដើម្បីភ្ជាប់ឲ្យបានចម្ងាយវែង យើងត្រូវប្រើ wireless router ឬ access point ។



យើងមានកុំព្យូទ័របីហើយបំពាក់ដោយ wireless adapter ។ អ្នកអាច setup ad hoc wireless នៅលើកុំព្យូទ័រមួយហើយចាត់ទុកថាជា host កុំព្យូទ័រ។ នេះគឺជា IP និង netmask ដែលយើងនិងកំណត់ឲ្យ wireless adapter នីមួយៗ។ សូមចំណាំថា អ្នកមិនចាំបាច់ត្រូវការកំណត់ gateway IP និង DNS ទេពីព្រោះថាវាជា peer-to-peer ណោតវើក ។

Host កុំព្យូទ័រ:

IP Address: 192.168.0.1

Subnet Mask: 255.255.255.0

Gateway:

DNS Servers:

Client កុំព្យូទ័រ A:

IP Address: 192.168.0.2

Subnet mask: 255.255.255.0

Gateway:

DNS Servers:

Client កុំព្យូទ័រ B:

IP Address: 192.168.0.3

Subnet mask: 255.255.255.0

Gateway:

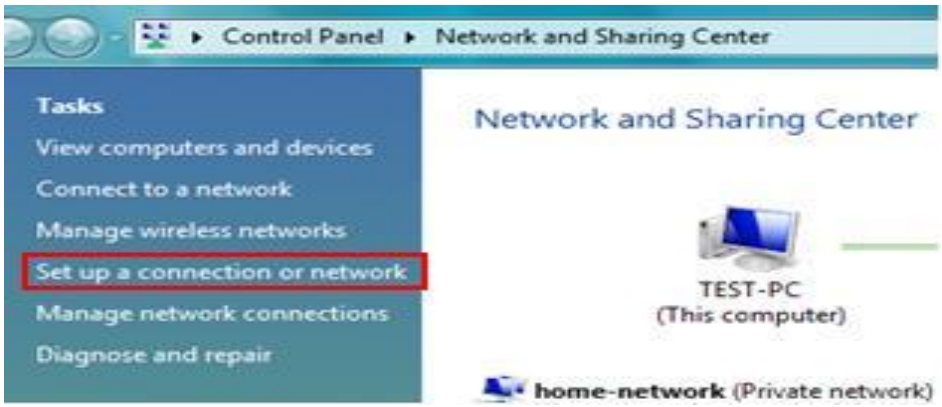
DNS Servers:

១-៨-៧-Host កុំព្យូទ័រ Configuration

នេះគឺជារបៀបនៃការ configure ad hoc wireless នៅលើ host កុំព្យូទ័រ។ ដូច្នេះកុំព្យូទ័រផ្សេងទៀតអាចភ្ជាប់ជាមួយវាតាម wireless ដោយផ្ទាល់ ។

១-មក **Start** ហើយចុចដោយប្រើប៊ូតុងMouse ខាងស្តាំលើ **ណេតវើក** ហើយបន្ទាប់មកចុចលើ **Properties**.

២- ណេតវើកand Sharing Center window will appear, click on **Set up a connection or network. Note**: The other way to do it is by going to **Start -> Connect To -> click on Set up a connection or network**.



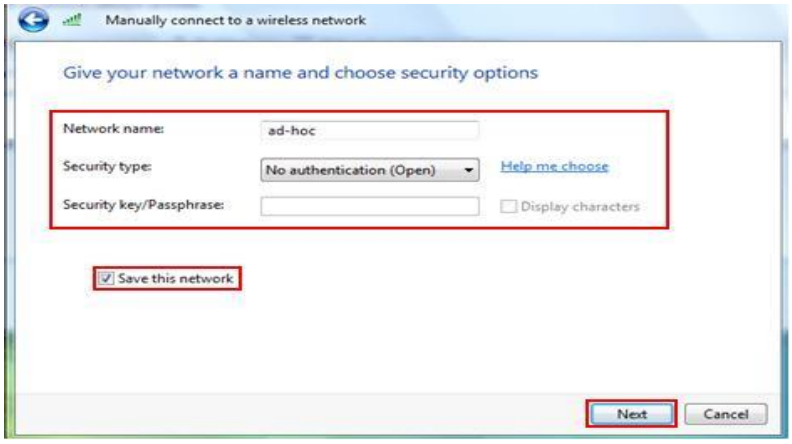
៣-Set up a connection or ណេតវើកwindow will appear, select **Set up a wireless ad hoc (កុំព្យូទ័រមកកាន់កុំព្យូទ័រ) ណេតវើក**option, then click **Next**

.

៤-បង្កើត wireless ad hoc ណេតវើកwindow មួយនិងបង្ហាញឡើងវានិងពន្យល់ឲ្យយើងដឹងពី ad hoc wireless ណេតវើក។ ចុចលើ **Next** ក្រោយពីអានចប់ ។

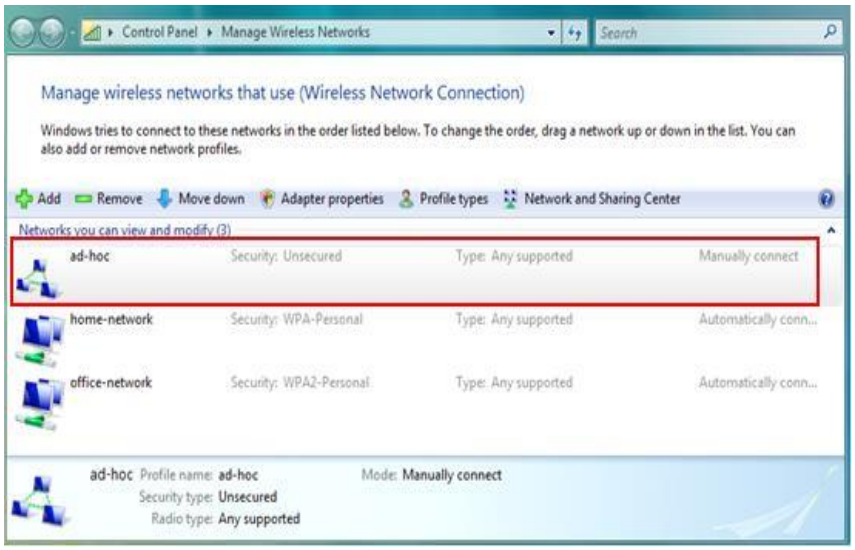
៥-នៅលើផ្ទាំង wireless ណេតវើកconfiguration window អ្នកអាចវាយបញ្ចូលឈ្មោះរបស់ ណេតវើកname (SSID), security type (encryption) និង security key អ្នកក៏អាចបង្កើត ad hoc connection ដោយមិនចាំបាច់ត្រូវការ authentication ក៏បានដែរហើយបន្ទាប់មកអាចបើក encryption (WEP,WPA2, etc) នៅពេលវាដំណើរការ។ សូមកុំភ្លេចបើក encryption ក្នុងករណីដែលគ្មានកុំព្យូទ័រនៅក្បែរនោះអាចភ្ជាប់ជាមួយ ណេតវើកនេះ ។ ចុចលើ **Next** ម្តងទៀត ។

ចំណាំ: អ្នកអាចចុចលើ **Save ណេតវើក**ដើម្បី save វាជា wireless ណេតវើកprofile។ បើមិនបាន Save វាទេ អ្នកនិងត្រូវបង្កើតវាម្តងទៀតនៅពេលអ្នកត្រូវការវា ។



៦- ផ្ទាំង window បន្ទាប់នឹងប្រាប់អ្នកអំពី ad hoc ណេតវើកត្រូវបានរួចរាល់សម្រាប់ប្រើ ចុចលើ **Close** ដើម្បីបញ្ចប់ការ setup ។

៧-ត្រឡប់មក ណេតវើកand Sharing Center ហើយចុចលើ **Manage Wireless networks**,អ្នកនិងឃើញ ad-hoc ណេតវើកដែលទើបបង្កើតថ្មីនិងវាបានភ្ជាប់រួចជាស្រេច ។



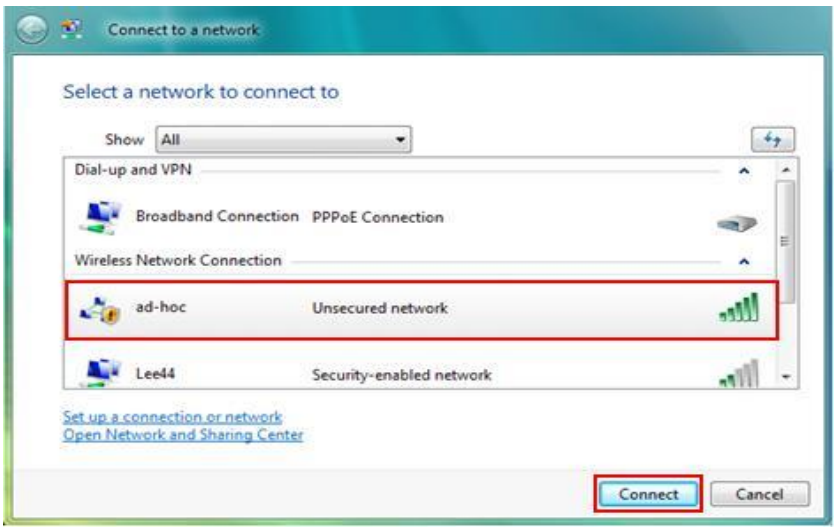
ចំណាំ:សូមចំណាំថា ad-hoc នេះគឺជា **Unsecured** ពីព្រោះវាគ្មាន encryption ត្រូវបានបើកនោះទេ។ ដូច្នេះកុំភ្លេចបើក encryption នៅពេលវាបានដំណើរការភ្លាម ។

១-៨-៨-Client កុំព្យូទ័រ Configuration:

បន្ទាប់មកអ្នកអាចប្រើ client កុំព្យូទ័រ ផ្សេងដើម្បីភ្ជាប់ជាមួយ ad hoc wireless ណេតវើក

1) ចុចលើ **Start** ហើយបន្ទាប់មកចុចលើ **Connect to**.

2) Connect to a ណេតវើកwindow និងបង្ហាញឡើង។ ផ្ទាំង window នេះនិងបង្ហាញពីការភ្ជាប់ដែលមានបម្រុងទុកជាស្រេចគឺ dial-up, VPN និង wireless ប៉ុន្តែអ្វីដែលយើងចាប់អារម្មណ៍គឺ ad hoc wireless



ចំណាំ: វានិងសួរអ្នកបើអ្នកភ្ជាប់មកកាន់ wireless ណេតវើកដែលគ្មានសុវត្ថិភាពហើយចុចលើ **Connect Anyway** ដើម្បីភ្ជាប់។

3) នៅពេលដែលបានភ្ជាប់ភ្លាម អ្នកនិងឃើញវាបង្ហាញ message។ អ្នកអាចចុចលើ **Save this ណេតវើក**ហើយបន្ទាប់មកចុចលើ Close the window ។ បន្ទាប់មកអ្នកអាចធ្វើការ test ដោយប្រើបញ្ជា Ping ។

១-៨-៩-របៀបការពារ Vista ពីការភ្ជាប់ជាមួយ Ad Hoc Wireless ណេតវើក

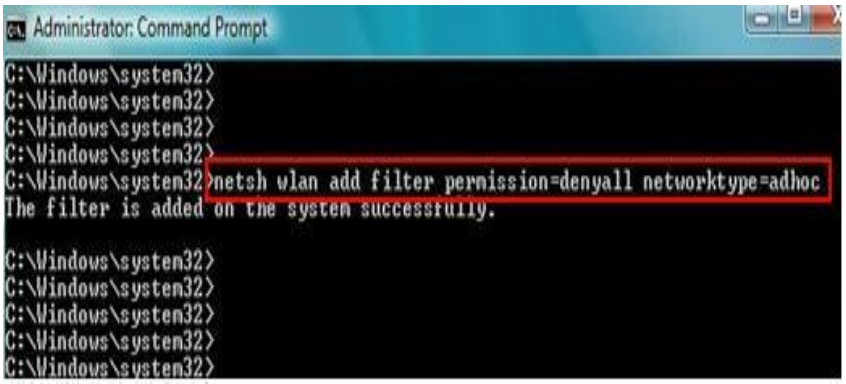
នេះគឺជាជំហានដ៏ខ្លីបំផុតសម្រាប់ច្រោះ ad hoc wireless ណេតវើកពីការភ្ជាប់:

1) Go to **Start -> All Programs -> Accessories** -> then right click on **Command Prompt (cmd.exe)** and click **Run as administrator**.

2) Command prompt window will និងបើកបង្ហាញឡើងហើយបន្ទាប់មកវាយបញ្ជា **netsh wlan show filters** ហើយចុចលើ **Enter** button ដើម្បីបង្ហាញពី wireless filter setting ដែលមាននៅលើកុំព្យូទ័រនេះ។

3) អ្នកអាចវាយបញ្ជា **netsh wlan add filter permission=denyall networktype=adhoc** ដើម្បីបិទ Ad hoc ណេតវើកពីការភ្ជាប់ ។

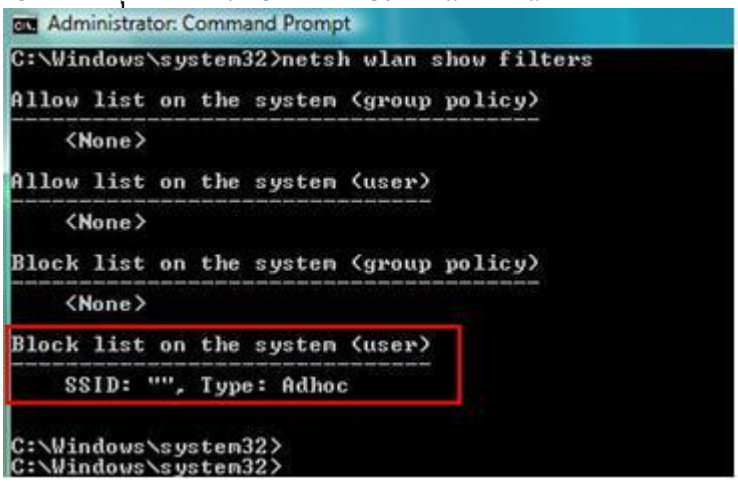
ចំណាំថាបើអ្នកឃើញ **Error: function WlanSetFilterList returns 5. The reques-ted operation requires elevation.** ក្រោយពីការប្រើបញ្ជាខាងលើ ។ អ្នកមិនអាចដំណើរការបញ្ជានេះបានទេពីព្រោះថាវាជាបញ្ជារបស់ administrator ។



4) បើអ្នកវាយបញ្ជា **netsh wlan show filters** ម្តងទៀតនោះអ្នកនិងឃើញវាដែលអ្នកបានកំណត់ឲ្យ។ អ្នកបានធ្វើវា កុំព្យូទ័រ និងមិនភ្ជាប់ជាមួយ ad hoc wireless ណេតវើកណាមួយនោះទេ

ចំណាំ:

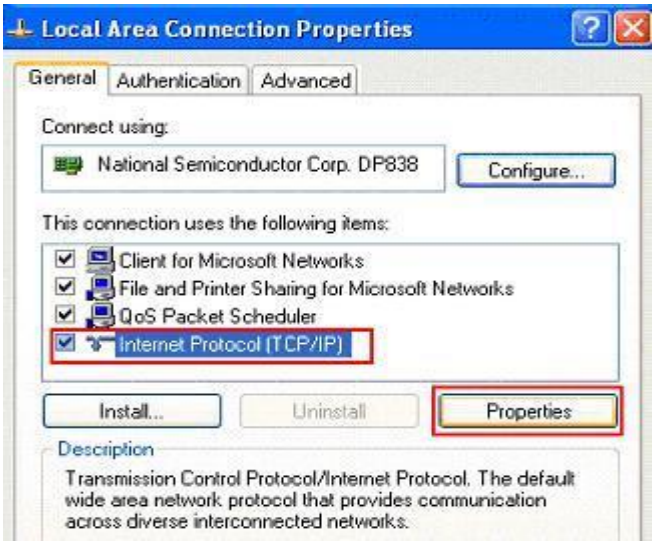
អ្នកអាចវាយបញ្ជា **netsh wlan delete filter permission=denyall networktype=adhoc** ដើម្បីលុបការប្រោះនេះក្នុងករណីអ្នកប្រាកដជាត្រូវការភ្ជាប់ជាមួយ ad hoc wireless ណេតវើក



១-៨-១០-របៀបកំណត់ IP Address និងព័ត៌មានអំពី ណេតវើកបន្ថែមទៀតក្នុង Windows XP

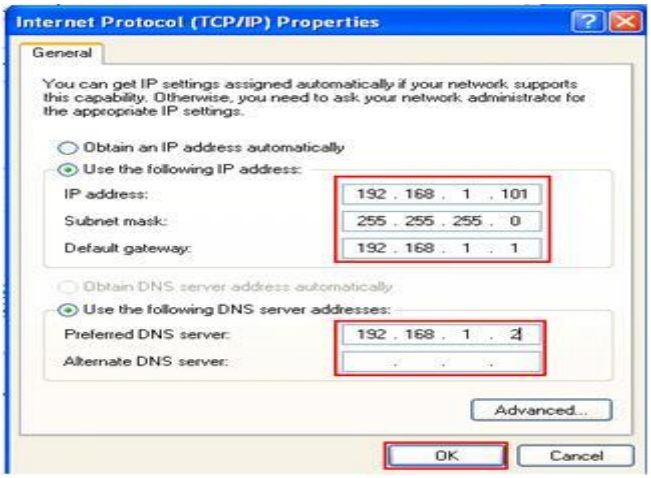
នេះគឺជាការណែនាំតាមជំហានៗនៃការកំណត់ IP address និងព័ត៌មានអំពី ណេតវើកបន្ថែមទៀត

1) Go to **Start** and click on **Control Panel**.
2) Control Panel window will appear. Double click on **ណេតវើកConnections**.
3) ណេតវើកConnections window will appear. Right click correct **Local Area Connection** by identifying correct ណេតវើកcard and click **Properties**.
4) Select **Internet Protocol (TCP/IP)**. Click on **Properties**.



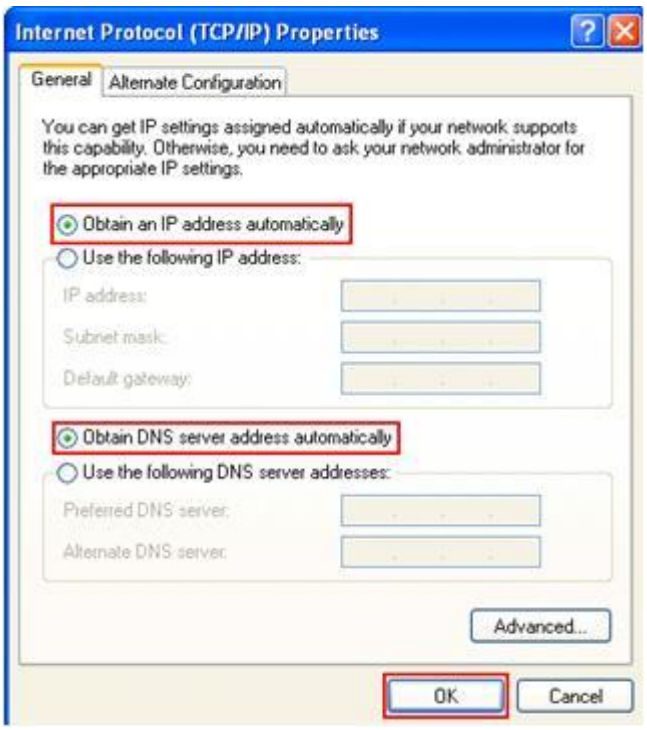
១-៨-១១-ការកំណត់ IP Address ដោយ ណេតវើកAdministrator

ចំណាំ: IP address របស់កុំព្យូទ័រត្រូវមានតែមួយគត់។ Default gateway គឺជា router មួយដែលអាច route ចរាចរណ៍ឲ្យ ណេតវើកផ្សេងទៀតឬ Internet។ DNS server គឺជា application server មួយដែលអាចបកប្រែ URL ឲ្យត្រូវជាមួយ IP address។ ឧទាហរណ៍ www.cert.org គឺជា URL ហើយវាត្រូវបានបំប្លែងឲ្យត្រូវជាមួយ 192.88.209.6 ដោយ DNS server ។



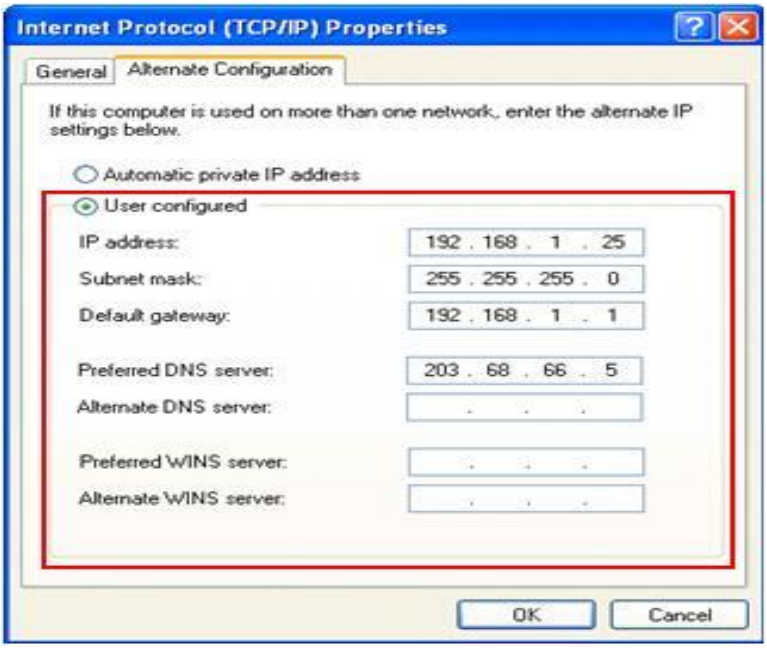
១-៨-១២-ការកំណត់ IP address តាមរយៈ DHCP server

បើអ្នកមាន DHCP server ដែលបាន setup នៅលើ router ឬអ្នកមាន DHCP server ក្នុង Home network កុំព្យូទ័ររបស់អ្នកអាចទទួលបាននូវ IP address ជាស្វ័យប្រវត្តដោយការជ្រើសរើសយក **Obtain an IP address automatically** និង **Obtain DNS server address automatically** ។



ចំណាំថាបើអ្នកមាន notebookមួយដែលកំពុងប្រើ static IP នៅឯផ្ទះហើយ IP ត្រូវបានកំណត់ឲ្យដោយ DHCP server នៅឯការិយាល័យ អ្នកអាចប្រើ **alternate configuration** ដើម្បីកំណត់ IP និងព័ត៌មានអំពី ណេតវើក សម្រាប់ network ពីរផ្សេងគ្នា។

ដើម្បីទទួលបាន IP address ជាស្វ័យប្រវត្តគឺនៅលើ **General** tab ដូចយើងបានកំណត់ខាងដើម។ ដូច្នេះ notebook និងត្រូវបានកំណត់ IP addresses ឲ្យជាស្វ័យប្រវត្តនៅឯការិយាល័យ។ ក្រោយពីនោះមកចុចលើ **Alternate Configuration** tab, ជ្រើសរើសយក **User configured** option និង key ក្នុង home network's static IP information ។



សំណួរត្រួតពិនិត្យ

១-ដូចម្តេចហៅថា Wirelesss LAN ?

២-តើ Wireless LAN មានគុណសម្បត្តិនិងគុណវិបត្តិអ្វីខ្លះ?

៣-ចូរបិទ Ad-hoc ណេតវើក

៤-ចូរបង្ហាញពីរបៀបបើក Ad-hoc network

៥-តើគេអាចសំគាល់ដឹងថា កុំព្យូទ័រ មាន Ad-hoc ឬ Acccess point Connection បានដោយសំគាល់លើអ្វី?

៦-តើ Wireless Router មានតួនាទីជាអ្វីខ្លះ?

៧-តើ Accesss Point មួយអាចភ្ជាប់បានចម្ងាយប៉ុន្មានម៉ែត្រ?

១-៩-ការណែនាំពីរបៀបប្រើខ្សែ

ដើម្បីគូសខ្សែគេត្រូវអនុវត្តតាមវិធីសាស្ត្រដូចខាងក្រោម:

- គេត្រូវប្រើខ្សែឲ្យបានច្រើនជាងចំនួនខ្សែដែលអ្នកត្រូវការពិតប្រាកដ
- យើងត្រូវការធ្វើតេស្តទៅលើខ្សែដែលយើងបានតវារួចហើយ
- ត្រូវដាក់វាឲ្យឆ្ងាយពីអំពូលភ្លើងនឹងប្រភពនសញ្ញាដែលជា Noise
- បើសិនជាក្នុងករណីដែលយើងត្រូវការអូសតាមជាន់នៃអាគារ
    - យើងត្រូវដាក់សបកឲ្យវាដើម្បីការពារចំពោះ Noise signal
- យើងត្រូវដាក់ស្លាកសញ្ញាសម្គាល់ឲ្យវាហើយត្រូវចងវាជាដុំដើម្បីដាក់នៅទីតាំងជាមួយគ្នា

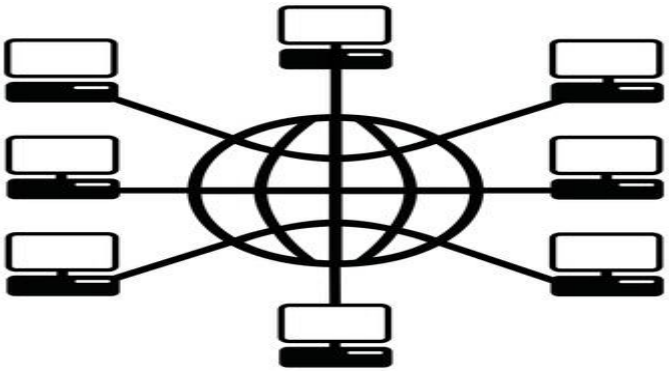
១-១០-តើ Topology ជាអ្វី?

រូបរាងនៃ ណេតវើកគឺសំដៅទៅលើការតេខ្សែកុំព្យូទ័រនិង Pheriherals ផ្សេងៗទៀត។ រូបរាង(Topplogy)មិនអាចច្រឡំជាមួយ Logical topology ដែលជាវិធីសាស្ត្រនៃការបញ្ជូនព័ត៌មានរវាង Workstations ។ Logical topology ត្រូវពិភក្សាជាមួយ Protocol ។

ប្រភេទនៃ Physical Topology សំខាន់ៗ

ផ្នែកខាងក្រោមនេះពិភាក្សាទៅលើរូបរាងដែលត្រូវបានប្រើនៅក្នុង ណេតវើកនិងអ្វីៗផ្សេងៗទៀតដែលទាក់ទង

- Linear Bus
- Star
- Star-wired Ring



កុំព្យូទ័រ ណេតវើកមួយ (local area ណេតវើកឬ wide area network) ដែលជាបន្សំនៃ កុំព្យូទ័រ,cablesនិង peripherals ផ្សេងៗទៀតដែលមានគោលបំណងសម្រាប់ចែកចាយទិន្នន័យនិង/ឬធនធាន គ្រោងឆ្អឹងនៃ ណេតវើកមួយគេហៅថាជា topology របស់វា។ វាមាន physical topologies ជាច្រើនប្រភេទដែលប្រភេទនីមួយៗមានគុណសម្បត្តិនិងគុណវិបត្តិរបស់វា។

១-១១-ងាយស្រួលដំឡើង មើលថែទាំនិងដោះស្រាយបញ្ហានៅពេលជួបបញ្ហា

វាមានលក្ខណៈងាយស្រួលក្នុងការបន្ថែម workstationsនិង peripherals ក្នុង star-wired ring topology។ ពីព្រោះថាគ្រប់ nodes ទាំងអស់ត្រូវបានភ្ជាប់ជាមួយ central hub ដោយផ្ទាល់ហើយគ្មានការជ្រៀតជ្រែកពីគ្នាទៅវិញទៅមកចំពោះការបន្ថែមឬការដក Node ណាមួយក្នុងប្រព័ន្ធ network។

១-១២-ងាយស្រួលប្រើខ្សែ

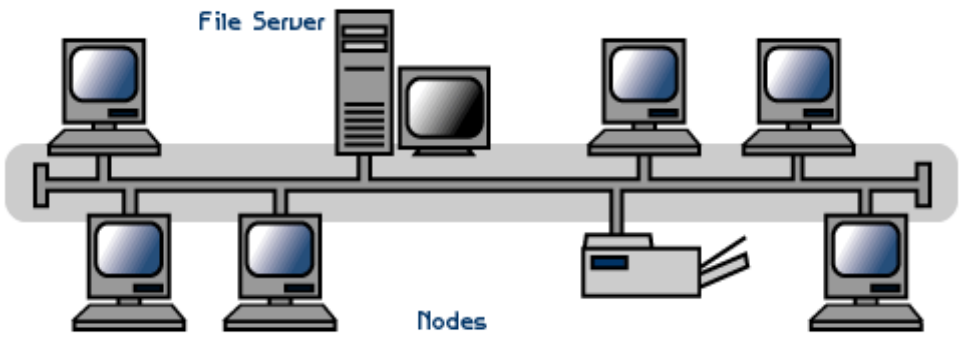
star-wired ring ណេតវើកមានសមត្ថភាពអាចភ្ជាប់ជាមួយខ្សែជាច្រើនប្រភេទ។ គេអាចប្រើខ្សែដូចជា fiber optic cabling, shielded twisted pair (STP) cables និង unshielded twisted pair (UTP)។ ប្រភេទនៃខ្សែដែលយកមកប្រើអាស្រ័យលើទំហំរបស់ ណេតវើកនិងតំរូវការចាំបាច់។

១-១៣-វាបានផ្តល់ឲ្យនូវ Fault Tolerance និងទំនុកចិត្តខ្ពស់

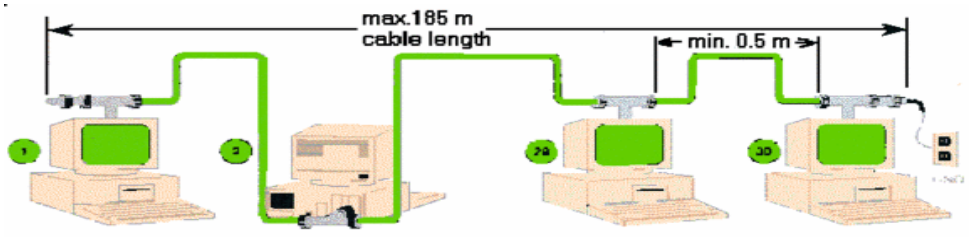
star-wired ring topology ផ្តល់ឲ្យនូវទិន្នន័យ fault tolerance ដ៏មានប្រសិទ្ធភាពខ្ពស់ជាង star topology Broadcast data ត្រូវបានទទួលបកប្រែនិងបញ្ជូនបន្តឲ្យ central hub។ access point នីមួយៗនៅលើណេតវើកគ្មានលក្ខណៈជា forwarding node(repeater)ដែលជាហេតុធ្វើឲ្យ មាន Error ក្នុងការបញ្ជូនទិន្នន័យ។ ក្រោយពីទិន្នន័យត្រូវបានទទួលនិងបកប្រែដោយ hub វាក៏បញ្ជូនបន្តឲ្យ Node បន្ទាប់នៅលើ network ហើយបន្ទាប់មកវាដំណើរការ។ បើ node មួយខូចឬខ្សែខូច hub នឹងបញ្ជូនបន្តទិន្នន័យឲ្យ Node បន្ទាប់ជាស្វ័យប្រវត្តិ។ មានន័យថាវាបង្កើនទំនុកចិត្តឲ្យប្រព័ន្ធ Network ដើម្បីជានាថាការបញ្ជូនទិន្នន័យមិនត្រូវបានប៉ះពាល់ឡើយ ។

១-១៤-Linear Bus

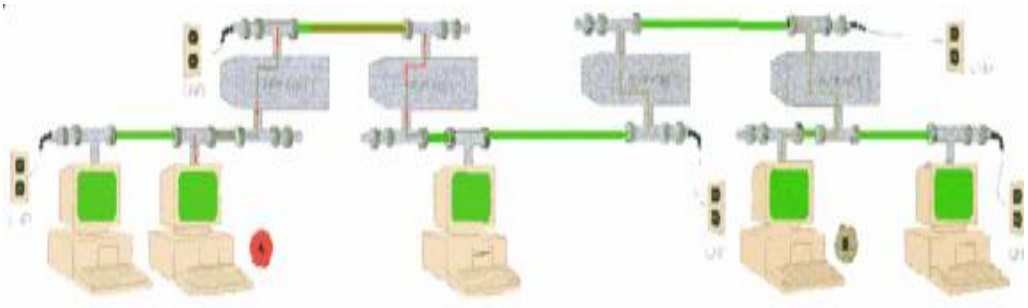
Linear bus topology រួមមានខ្សែរមួយភ្ជាប់ជាមួយ terminator គ្រប់នៅចុងទាំងសងខាង។ គ្រប់ Node ទាំងអស់ (Fileserver, Workstation និង Pheriperals) ត្រូវតែភ្ជាប់ទៅនិង Linear cable។ Ethernet និង LocalTalk ណេតវើកប្រើ Linear bus topology ។



ការភ្ជាប់ដោយប្រើ 10 base 2



ការភ្ជាប់ដោយប្រើ 10 base 5



គុណសម្បត្តិនៃ Linear Bus Topology

- ងាយស្រួលក្នុងភ្ជាប់កុំព្យូទ័រឬ Peripherals ទៅនិង Linear Bus
    - ត្រូវការប្រវែងខ្សែតិចជាង Star topology

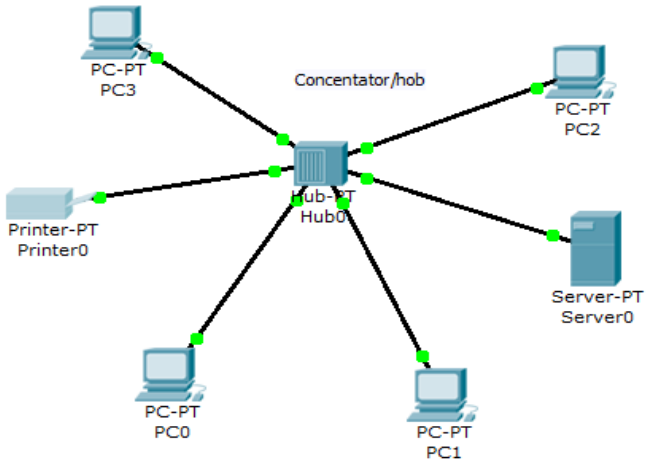
គុណវិបត្តិនៃ Linear Bus Topology

- ប្រព័ន្ធទាំងមូលខូច បើខ្សែមេខូច
- Terminator ត្រូវការសម្រាប់សងខាងទាំងពីរនៃខ្សែមេ
- ពិបាកេកបញ្ហាប្រសិនបើប្រព័ន្ធ ណេតវើកទាំងមូល
    - មិនមែនមានន័យថាវាត្រូវបានប្រើជាដំណោះស្រាយនៅក្នុងអាគារដែលមានទំហំធំនោះទេ

១-១៥-Star

Star topology ត្រូវបានរៀបចំឡើងជាមួយគ្រប់Nodes (Fileserver, Workstation និង peripherals) ដើម្បីភ្ជាប់ផ្ទាល់ទៅកាន់ Hub/Switch ឬ Concentator ដែលជាចំណុចកណ្តាល។

ទិន្នន័យនៅលើ Star ណេតវើកត្រូវបានបញ្ជូនឆ្លងកាត់តាម Hub ឬ Concentrator មុនពេលបញ្ជូនបន្តទៅកាន់គោលដៅរបស់វា។ ការ Configure នេះជាទូទៅត្រូវបានប្រើជាមួយខ្សែ UTP ថ្វីបើវាអាចប្រើជាមួយ Caoxial ឬ Fiber optic ក៏ដោយ។



គុណសម្បត្តិនៃ Star Topology

- ងាយស្រួលក្នុងការដំឡើងនិងតភ្ជាប់
  - គ្មានការបង្អាក់ចំពោះ ណូតរើកបន្ទាប់ពីការភ្ជាប់ឬការដកយកឧបករណ៍ចេញនោះទេ
  - ងាយស្រួលក្នុងការស្វែងរកំហុសនិងដោះស្រាយយកផ្នែកណាមួយចេញ

គុណវិបត្តិនៃ Star Topology

- ត្រូវការខ្សែច្រើនជាង Linear Bus topology
- ប្រសិនបើ Hub ឬ Concentrator ខូច នោះ Node ដែលបានភ្ជាប់គឺមិនអាចប្រើប្រាស់បាននោះទេ
- មានតម្លៃថ្លៃជា Linear Bus ពីព្រោះតម្លៃនៃ Concentrator ថ្លៃ

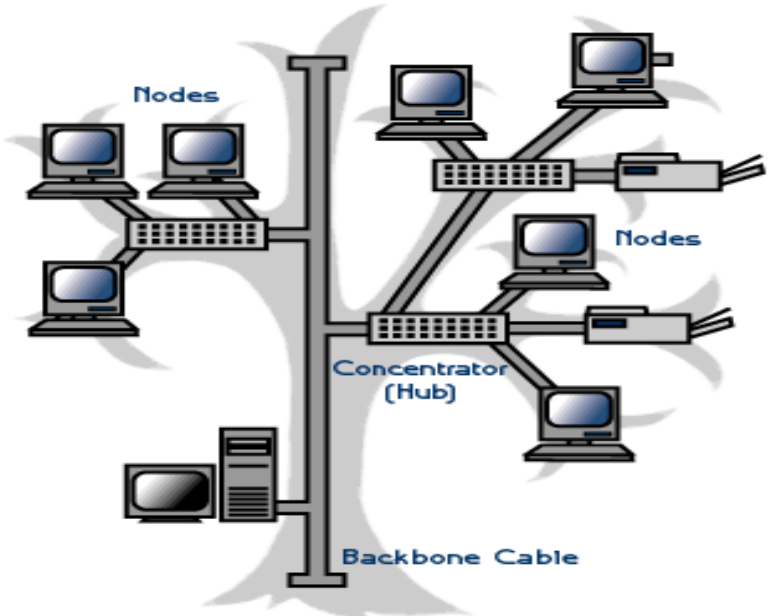
Protocols ត្រូវបានប្រើជាមួយ Star ឬ Ethernet ឬ localTalk។ Token ត្រូវបានគេប្រើជាមួយ Topology ដែលគេហៅថា Star-Wired Ring ។

១-១៦-Star-Wired Ring

Star-wired ring topology អាចលេចឡើងមានរាងដូចជា Star topology ដែរ។ MAU (Multistation Access Unit) នៃ Star-Wired Ring មានខ្សែដែលអនុញ្ញាតឲ្យព័ត៌មានឆ្លងកាត់ពីឧបករណ៍មួយទៅឧបករណ៍មួយទៀតនៅក្នុងរង្វង់។ Token Ring protocol ប្រើ Star-Wired Ring ។

១-១៧-Tree

Tree topology មានលក្ខណៈបញ្ចូលគ្នានៃ Linear Bus និង Star topology ។ វារួមមានក្រុមនៃ Star Workstation ភ្ជាប់ទៅនិង Linear Bus backbone ។ Tree Topology អនុញ្ញាតឲ្យអាចពង្រីក ណូតរើកដែលមានស្រាប់និងអាចឲ្យគេ Configure ចំពោះ ណូតរើកទៅតាមតម្រូវការរបស់គេ។



គុណសម្បត្តិនៃ Topology

- សម្រាប់ភ្ជាប់ពីចំណុចមួយទៅចំណុចមួយទៀតសម្រាប់ Segment នីមួយៗ
- មានការគាំទ្រដោយ Hardware និង Sofware ខ្លះ

គុណវិបត្តិនៃ Tree topology

- ប្រវែងនៃខ្សែនៃគ្រប់ Segment ត្រូវបានកំនត់ដោយប្រភេទនៃខ្សែដែលបានប្រើ
    - ប្រសិនបើខ្សែ Backbone ខូច នោះ Segment ទាំងមូលនឹងខូចដែរ
    - ពិបាកក្នុងការ Configure នឹងតខ្សែជាង topology ផ្សេងៗទៀត

១-១៨-ច្បាប់ 5-4-3

គួរពិចារណាក្នុងការដំឡើងនៃ Tree topology ដោយប្រើ Ethernet Protocol គឺច្បាប់៥-៤-៣។ ទិដ្ឋភាពមួយនៃតម្រូវការ Ethernet protocol គឺសញ្ញាលបានបញ្ជូនចេញទៅក្រៅតាមខ្សែ ណេតវើកទៅដល់គ្រប់ផ្នែកនៃ Nerwork ជាមួយរយៈពេលច្បាស់លាស។ គ្រប់ Concentrator ឬ Repeater ដែលសញ្ញាលឆ្លងកាត់ត្រូវបន្ថែមចំនួននៃពេលវេលាតូចមួយ។ នេះគឺជាគោលការណ៍កំនត់រវាង node ពីរនៅលើ ណេតវើកដែលមានប្រវែងត្រឹម៥កង់ដែលអាចភ្ជាប់បាន 4 repeaters/Concentrator ។ បន្ថែម ៣កង់ទៀតអាចធ្វើឲ្យប្រើប្រាស់ភ្ជាប់ Workstatiob ដើមប្រសិនបើវាត្រូវបានប្រើជាមួយខ្សែ Coaxial ។

Segment ដើមគឺជា Segment មួយដែល node ពីរឬច្រើនភ្ជាប់ជាមួយវា។ Node ពីរដែលនៅឆ្ងាយនៅលើ **ណេតវើក**មាន 4 segments និង 3 repeater/concentrator រវាងពួកវា។

នេះគឺជាច្បាប់ដែលមិនអាចអនុវត្តបានចំពោះ **ណេតវើក**protocol ផ្សេងទៀតឬ Ethernet **ណេតវើក**ដែលគ្រប់ Fiber Optic ឬបន្សំនៃ Fiber backbone ជាមួយខ្សែ UTP ។ ប្រសិនបើបន្សំនៃមេ Fiber optic និងខ្សែ UTP ត្រូវប្រើច្បាប់ ៧-៦-៥។

គួរពិចារណានៅពេលជ្រើសរើសយក Topology

- Linear Bus **ណេតវើក**អាចមានតម្លៃថោក។វិធីនៃការដំឡើង **ណេតវើក**មិនចាំបាច់ទិញ Concentrator នោះទេ
- ប្រវែងនៃខ្សែដែលត្រូវការសម្រាប់ **ណេតវើក**គឺប្រើប្រវែងខ្សែខ្លី
- ប្រភេទនៃខ្សែ:ភាគច្រើនគេប្រើ UTP ដែលត្រូវបានគេប្រើជាមួយ Star

តារាងសង្ខេប

Physical Topology	Common Cable	Common Protocol
Linear Bus	Twisted Pair Coaxial Fiber	Ethernet LocalTalk
Star	Twisted Pair Fiber	Ethernet LocalTalk
Star-Wired Ring	Twisted Pair	Token Ring
Tree	Twisted Pair Coaxial Fiber	Ethernet

១-១៩-តើ ណេតវើកOperationg System ជាអ្វី ?

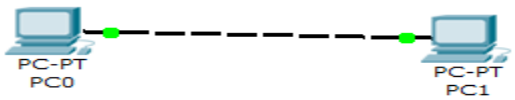
វាមិនដូចទៅនិងប្រព័ន្ធប្រតិបត្តិការដូចជា DOS និង Windows នោះទេដែលត្រូវបានដំឡើងដោយអ្នកប្រើប្រាស់តែម្នាក់គត់ដើម្បីគ្រប់គ្រងទៅលើកុំព្យូទ័រមួយ។ ប្រព័ន្ធប្រតិបត្តិការ **ណេតវើក**ធ្វើសកម្មភាពជាមួយកុំព្យូទ័រជាច្រើនដែលឆ្លងកាត់ប្រព័ន្ធ network។ ប្រព័ន្ធប្រតិបត្តិការ **ណេតវើក**ដើរតួនាទីជាប្រធាននៃដំណើរការរបស់ **ណេតវើក**ឲ្យបានស្រួល។ ប្រភេទនៃប្រព័ន្ធប្រតិបត្តិការ **ណេតវើក**គឺ:

- Peer-to-Peer
- Client/Server

១-១៩-១-Peer-to-Peer

Peer-to-Peer **ណេតវើក**operating system អនុញ្ញាតឲ្យអ្នកប្រើប្រាស់អាចចែកចាយធនធាននិង files ដែលស្ថិតនៅលើកុំព្យូទ័រដើម្បីដំណើរការចំពោះធនធាន ដែលបានចែកចាយឲ្យបានឃើញនៅលើកុំព្យូទ័រផ្សេងៗទៀត។ ទោះបីជាយ៉ាងណាក៏ដោយវាគ្មាន Fileserver ឬប្រភពកណ្តាលដើម្បីគ្រប់គ្រងនៅក្នុង Peer-to-Peer network គឺគ្រប់កុំព្យូទ័រទាំងអស់ត្រូវបានចាត់ទុកស្មើៗគ្នា។ វាមានសិទ្ធស្មើគ្នាចំពោះការប្រើប្រាស់ធនធានដែលមានស្រាប់នៅលើ network។ Peer-to-Peer network ត្រូវបានគេបង្កើតឡើងប្រើប្រាស់ភាគច្រើននៃ LAN ដែលមានទំហំតូច Appeshare នឹង Windows សម្រាប់workgroup គឺជាឧទាហរណ៍នៃកម្មវិធីដែលអាចដើរតួនាទីជា peer-to-peer **ណេតវើក**operating system។

Peer-to-Peer Network



គុណសម្បត្តិនៃ **Peer-to-Peer network**

- ចំណាយទុនតិចដោយមិនចាំបាច់ទិញ Server
- Windows ដូចជា Windows XP, Windows 8, Windows 10 ត្រូវបាន Configure ជាស្រេចសម្រាប់ Peer-to-Peer network

គុណវិបត្តិនៃ Peer-to-Peer network

- គ្មានកន្លែងកណ្តាលសម្រាប់ Files នឹង applications
- គ្មានសុវត្ថិភាព

១-១៩-២-Client/Server

DELL

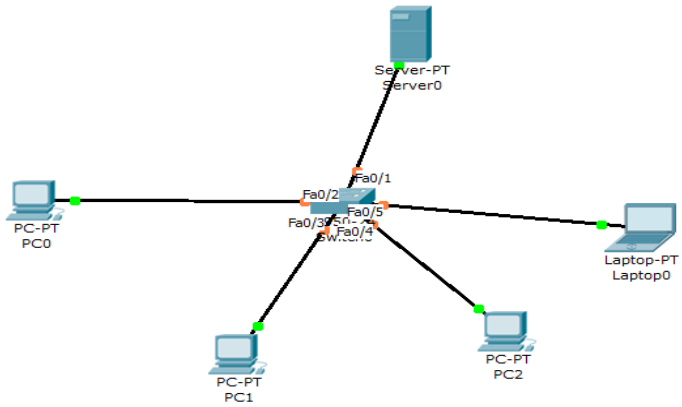
DELL

Blade Servers

Rack system

អនុញ្ញាតឲ្យអ្នកប្រើប្រាស់អាចគ្រប់គ្រងទៅលើ applications ដែលស្ថិតនៅលើ Fileserver។ ដូច្នេះ Fileserver គឺជាកុំព្យូទ័រដ៏សំខាន់មួយនៅក្នុងប្រព័ន្ធ **ណេតវើក**ពីព្រោះថាវាអនុញ្ញាតឲ្យយើងអាចប្រើប្រាស់បាននូវធនធាននឹងផ្តល់នូវសុវត្ថិភាព។

Workstation ឬ clientនីមួយៗអាចប្រើប្រាស់នូវធនធានទាំងអស់ដែលមាននៅលើ Fileserver ។ ប្រព័ន្ធប្រតិបត្តិការ **ណេតវើក**អនុញ្ញាតឲ្យយើងបញ្ចូលនូវសមាសធាតុទាំងអស់បញ្ចូលគ្នាហើយអនុញ្ញាតឲ្យអ្នកប្រើប្រាស់អាចចែកចាយនូវធនធាននៅពេលព្រមគ្នាបាន។ Novell Netware និង Window 2000 server/2008/2012 /2016 គឺជាប្រព័ន្ធប្រតិបត្តិការនៃប្រព័ន្ធកូន-មេនៃប្រព័ន្ធ **ណេតវើក**។



គុណសម្បត្តិនៃប្រព័ន្ធ **ណេតវើក**មេ-កូន

- Centralized: គ្រប់ធនធាននិងសុវត្ថិភាពនៃទិន្នន័យត្រូវបានគ្រប់គ្រងដោយម៉ាស៊ីនមេ។ មានន័យថាការប្រើប្រាស់នៃធនធានបានឬមិនបានអាស្រ័យទៅលើម៉ាស៊ីនមេជាអ្នកកំនត់ឲ្យ។
- Scalability: គ្រប់សមាសធាតុរបស់ **ណេតវើក**គេអាចជំនួសឬដោះដូរបានទៅតាមតម្រូវការនៃការកើនឡើង
- Flexibility: គេអាចបញ្ចូលបាននូវបច្ចេកវិទ្យាថ្មី
- Interoperability: គ្រប់សមាសធាតុទាំងអស់ត្រូវតែដំណើរការជាមួយគ្នាបាន
- Accessibility: គេអាចគ្រប់គ្រងទៅលើម៉ាស៊ីនមេតាម Platforms

គុណវិបត្តិនៃប្រព័ន្ធ **ណេតវើក**មេ-កូន

- ចំណាយទុនច្រើនដើម្បីទិញម៉ាស៊ីនមេ
- ការថែទាំ:ចំពោះប្រព័ន្ធ **ណេតវើក**ធំត្រូវការបុគ្គលិកដែលមានជំនាញច្បាស់លាស់ដើម្បីជានាថា**ណេតវើក**មានប្រសិទ្ធភាពខ្ពស។
- ភាពអាស្រ័យគ្នា:បើម៉ាស៊ីនមេខូចឬមិនដំណើរការនោះប្រព័ន្ធ **ណេតវើក**ត្រូវបានគាំង

ឧទាហរណ៍នៃប្រព័ន្ធប្រតិបត្តិការ **ណេតវើក**មានដូចខាងក្រោម:

- AppleShare
- Microsoft Windows Server
- Novell Netware

សំណួរត្រួតពិនិត្យ

១-ចូរពន្យល់ច្បាប់ ៥-៤-៣ ?

២-ចូរពន្យល់ច្បាប់ ៧-៥-៦ ?

៣-តើ Peer to Peer ណេតវើកមានគុណសម្បត្តិនិងគុណវិបត្តិអ្វីខ្លះ ?

៤-តើ Client/Server ណេតវើកមានគុណសម្បត្តិនិងគុណវិបត្តិអ្វីខ្លះ ?

ជំពូកទី២

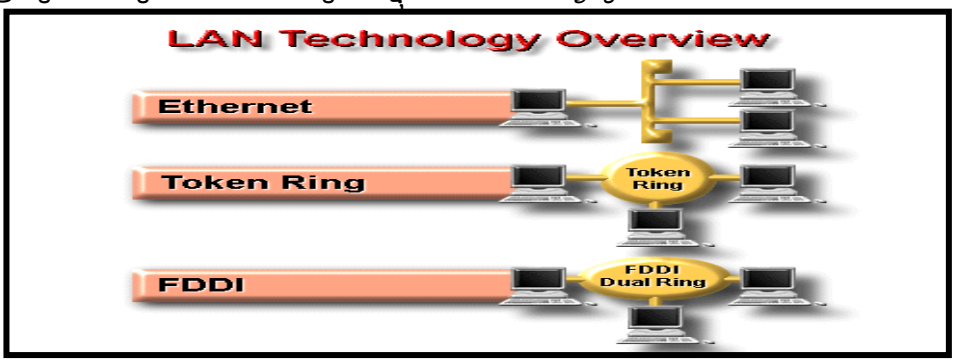
ការរម្លឹកឡើងវិញអំពី LAN Technology និង

ការរៀបចំគម្រោងនៃការបង្កើតប្រព័ន្ធ Network

មុននិងអ្នកសិក្សាព័ត៌មានច្បាស់លាស់អំពីបច្ចេកវិទ្យានៃ LAN ដែលគេនិយមប្រើ។ មានបច្ចេកវិទ្យានៃ LAN ចំនួន៣ដែលត្រូវបង្ហាញក្នុងគឺ:

- Ethernet: គឺជាបច្ចេកវិទ្យានៃ LAN ដែលសំខាន់ហើយវាជំណើរការនៅក្នុង LAN
- Token Ring: គឺជាបច្ចេកវិទ្យាដែលបានមកពីក្រុមហ៊ុន IBM ហើយឥឡូវនេះគេប្រើវាយ៉ាងច្រើននោះដែរ
- FDDI: គឺបានប្រើប្រាស់នូវ Token ផងដែរហើយសព្វថ្ងៃនេះគេពេញនិយមប្រើប្រាស់ជា LAN ដែលដំឡើងនៅតាមអាគារ។ យើងនិងសិក្សាពី Ethernet 802.3 ។

តាមរូបភាពបង្ហាញពីប្រភេទខ្សែដែលគេនិយមប្រើនៅក្នុង ណេតវើកសព្វថ្ងៃនេះគឺ Caxial Fiber optic និង UTP ។



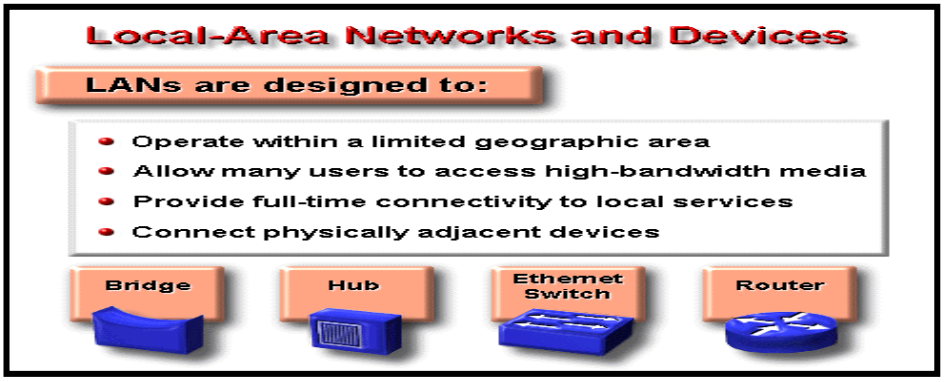
๒-๑-Local Area Networks and Devices

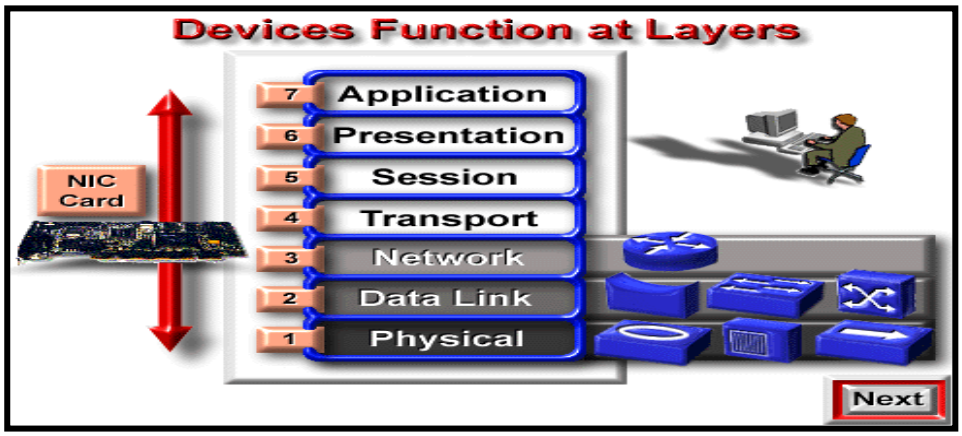
នេះគឺជាលក្ខណៈដ៏សំខាន់នៃ LAN:

- ជាប្រភេទបណ្តាញណេតវើកមួយដែលគេបានប្រើប្រាស់នៅក្នុងអាគារឬការិយាល័យនៃអាគារណាមួយ
- LAN បានអនុញ្ញាតឲ្យយើងអាចភ្ជាប់ឧបករណ៍ច្រើន (តាមធម្មតាជា PC) ដែលមានខ្សែមានល្បឿនលឿន។ បើតាមនិយមន័យនៃ LAN ត្រូវបានគេភ្ជាប់តាមរយៈ Physical layer ។

ឧបករណ៍របស់ LAN រួមមានដូចជា:

- Bridge: គឺត្រូវបានគេប្រើប្រាស់សម្រាប់ភ្ជាប់ទៅកាន់ LAN segment និងច្រោះចរាចរណ៍
- Hub: វាមានតួនាទីសម្រាប់តែភ្ជាប់ប៉ុណ្ណោះហើយអនុញ្ញាយឲ្យប្រើប្រាស់ខ្សែប្រភេទជា UTP
- Ethernet Switch: ត្រូវបានគេប្រើប្រាស់សម្រាប់បញ្ជូនទិន្នន័យដែលមានលក្ខណៈជា Full duplex ហើយអាចកំនត់នូវ bandwidth ទៅឲ្យ Segment ឬឧបករណ៍ណាមួយ
- Router: វាអាចផ្តល់ឲ្យនូវសេវាជាច្រើនរួមមាន Internetworking និងគ្រប់គ្រងទៅលើ broadcast

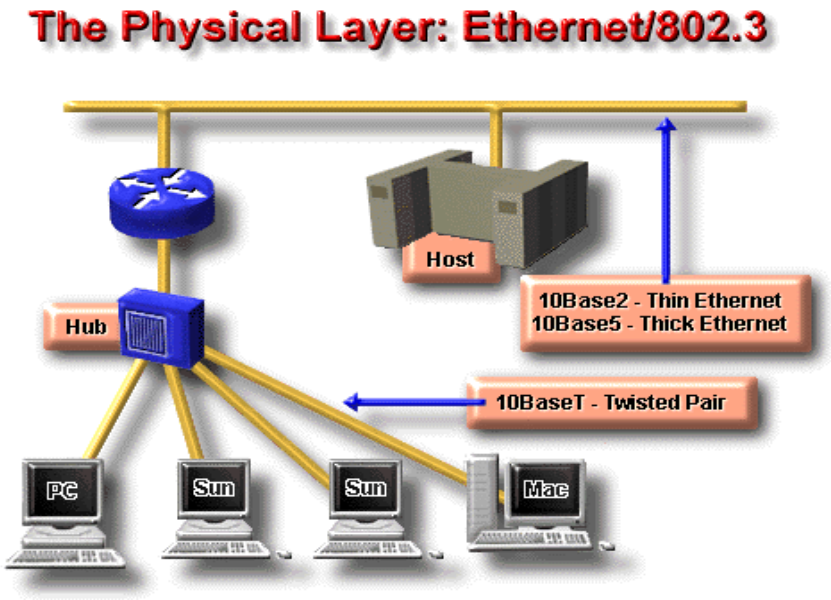




២-២-Physical Layer : Ethernet/802.3

Ethernet និង IEEE 802.3 standards បានកំនត់ពីរូបរាងនៃ LAN ជា Bus ដែលប្រើប្រាស់នូវ Baseband signaling ដែលមានអត្រាល្បឿន 10 Mbps។ តាមរូបបង្ហាញនូវស្តង់ដាបីប្រភេទនៃ LAN គឺ:

- 10 Base 2: គឺជា Thin Ethernet អនុញ្ញាតឲ្យប្រើប្រាស់ត្រឹមតែចម្ងាយ១៨៥ម៉ែត្រដែលជាខ្សែ Coaxial
- 10 Base 5: គឺជា Thick Ethernet អនុញ្ញាតឲ្យប្រើប្រាស់បានចម្ងាយរហូតដល់៥០០ម៉ែត្រដែលជា Coaxial
- 10 Base T:គឺជា Carrier Ethernet Frame វាមានតម្លៃមិនស៊ូវថ្លៃនោះទេ វាជាខ្សែ UTP ។ 10 base 5 និង10 base 2 ត្រូវបានផ្តល់នូវការប្រើសម្រាប់ក្នុង LAN segment ដូចគ្នា។ ឧបករណ៍ត្រូវបានគេភ្ជាប់ទៅនិង AUI (Attachment Unit Interface) តាមរយៈខ្សែទៅកាន់ Transceiver ដែលត្រូវបានភ្ជាប់ដោយផ្ទាល់ទៅកាន់ខ្សែ Ethernet Coaxial ។ ដោយសារតែ 10 base T ត្រូវបានគេប្រើដើម្បីប្រើសម្រាប់តែឧបករណ៍តែមួយ។ ឧបករណ៍ជាច្រើនត្រូវបានគេភ្ជាប់ទៅនិង Ethernet LAN ដោយប្រើខ្សែ 10 base T ដែលត្រូវបានភ្ជាប់ទៅ Hub ឬ LAN switch មួយ។ នៅក្នុងការរៀបចំនេះ Hub ឬ LAN switch ដូចគ្នាទៅនិង Ethernet segment ដែរ។

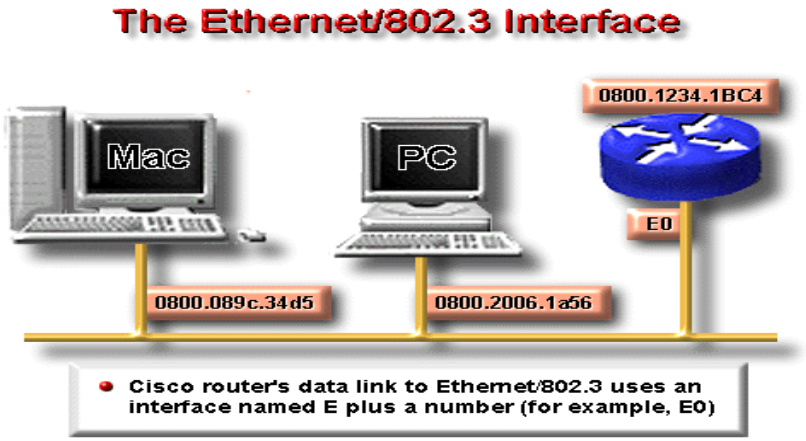


២-២-Ethernet/802.3 Interface

Ethernet និង 802.3 data link បានរៀបចំនូវទិន្នន័យសម្រាប់ការបញ្ជូនឆ្លងកាត់តាមខ្សែដែលត្រូវបានគេប្រើប្រាស់ដើម្បីភ្ជាប់ទៅឧបករណ៍ទាំងពីរ។

ឧទាហរណ៍

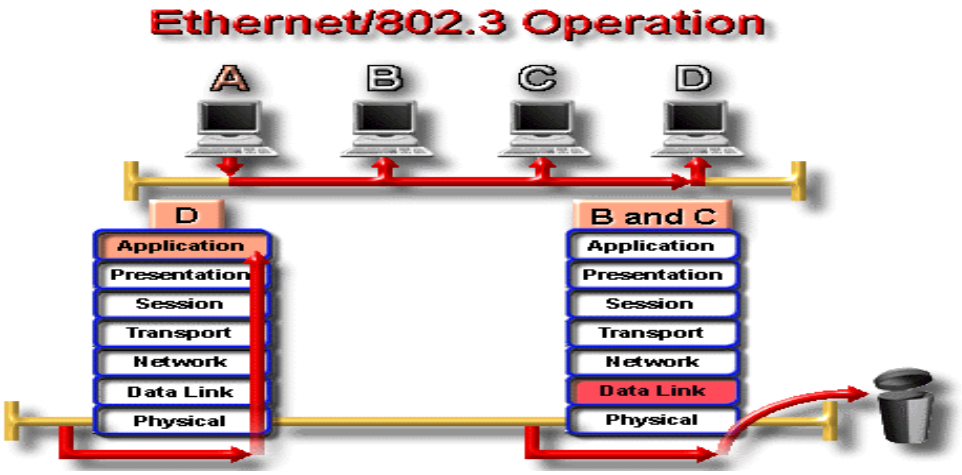
ដូចនៅក្នុងរូបភាពនេះបង្ហាញនូវឧបករណ៍ចំនួន៣ត្រូវបានគេតភ្ជាប់គ្នាទៅវិញទៅមកនៅលើ Ethernet LAN មួយ។ ម៉ាស៊ីនជាប្រភេទ Mancintosh នៅខាងឆ្វេងនិង Intel-based pc ស្ថិតនៅកណ្តាលបានបង្ហាញនូវ MAC address បានប្រើដោយ Datalink Layer។ Router នៅខាងស្តាំប្រើប្រាស់នូវ MAC address ផងដែរសម្រាប់ LAN interface នីមួយៗរបស់វា។ ចំពោះ Ethernet /802.3 interface នៅក្នុង router ដែលយើងបានប្រើប្រាស់ជាមួយ Cisco IOS interface ដែលតាងដោយអក្សរកាត់គឺ E ប្រើសម្រាប់លេខ Interface(ឧទាហរណ៍ដូចបង្ហាញក្នុងរូប E 0)។

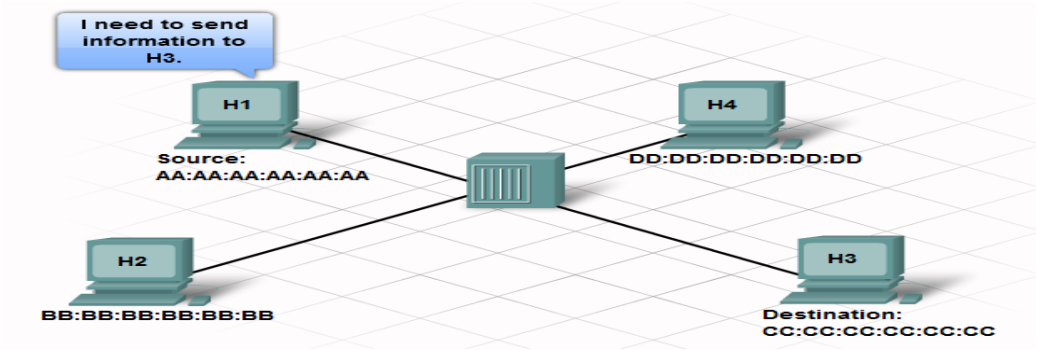


២-៣-Ethernet/802.3 Operation

នៅក្នុង Carrier Sense Multile Access ដែលមាននូវ Collision Detection(CSMA/CD) network កើតឡើងហើយការបញ្ជូនរបស់ Node មួយត្រូវបានឆ្លងកាត់ ណេតវើកទាំងមូលហើយវាត្រូវបានទទួលនិងព្រមទាំងត្រូវបានត្រួតពិនិត្យតាមគ្រប់ Node នីមួយៗ។ នៅពេលដែលសញ្ញាលមួយបានមកដល់ចុងបញ្ចប់នៃ Segement

ហើយនៅពេលនោះ Terminator បានស្រូបយកវាដើម្បីការពារវាមិនឲ្យវិលត្រឡប់មកវិញនៅលើ Segment ។ ដោយអនុញ្ញាតឲ្យមានការបញ្ជូនតែមួយគត់នៅលើ LAN មួយនៅពេលដែលគេបានផ្តល់ឲ្យ។

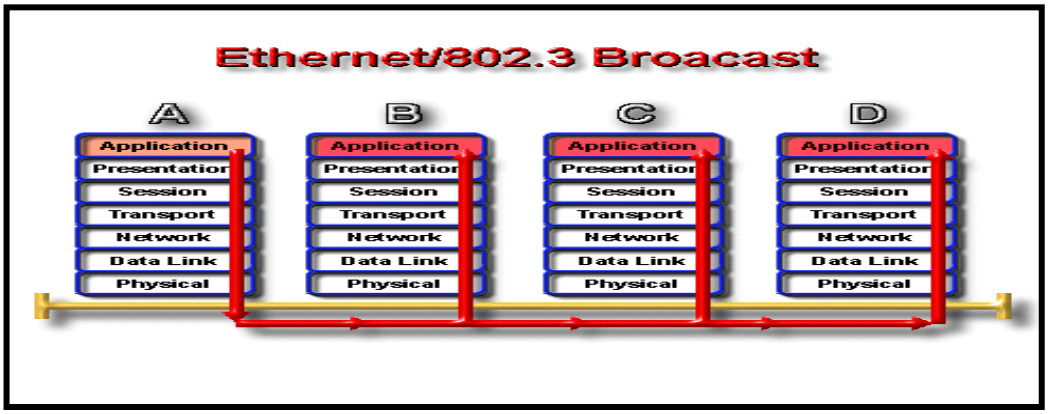




២-៤-Ethernet / 802.3 Broadcast

Broadcasting គឺជា Tool មួយដែលមានឥទ្ធិពលបំផុតដែលវាធ្វើការបញ្ជូននូវ Frame មួយទៅឲ្យឧបករណ៍ជាច្រើននៅពេលតែមួយ។ Broadcast បានប្រើប្រាស់នូវអាស័យដ្ឋានរបស់គោលដៅស្ថិតនៅក្នុង Datalink ដែលមានទម្រង់ជា bits លេខ១ទាំងអស់ (FFFFFFFFFFFFFFFF គោល១៦) ។ ដូចនៅក្នុងរូបបានបង្ហាញពីឧបករណ៍ A បានធ្វើការបញ្ជូននូវ Frame មួយដែលមាននូវអាស័យដ្ឋានរបស់គោលដៅរបស់ឧបករណ៍ទាំងអស់ហើយនៅពេលនោះឧបករណ៍ B C និង D បានទទួលហើយបានផ្តល់ទៅឲ្យស្រទាប់ខាងលើសម្រាប់ដំណើរការបន្តទៀត។

Broadcast បានធ្វើឲ្យប៉ះពាល់យ៉ាងខ្លាំងដល់ដំណើរការរបស់ប្រព័ន្ធ ណេតវើកព្រោះថាវាបានរំខានដល់ឧបករណ៍ដែលស្ថិតនៅលើប្រព័ន្ធ ណេតវើកនោះ។ ដូច្នេះ broadcast ត្រូវបានប្រើប្រាស់តែក្នុងករណីដែល MAC address របស់គោលដៅមិនត្រូវបានស្គាល់ឬនៅពេលដែលវាធ្វើការបញ្ជូននូវ frame មួយទៅឲ្យគ្រប់ឧបករណ៍ទាំងអស់ ។



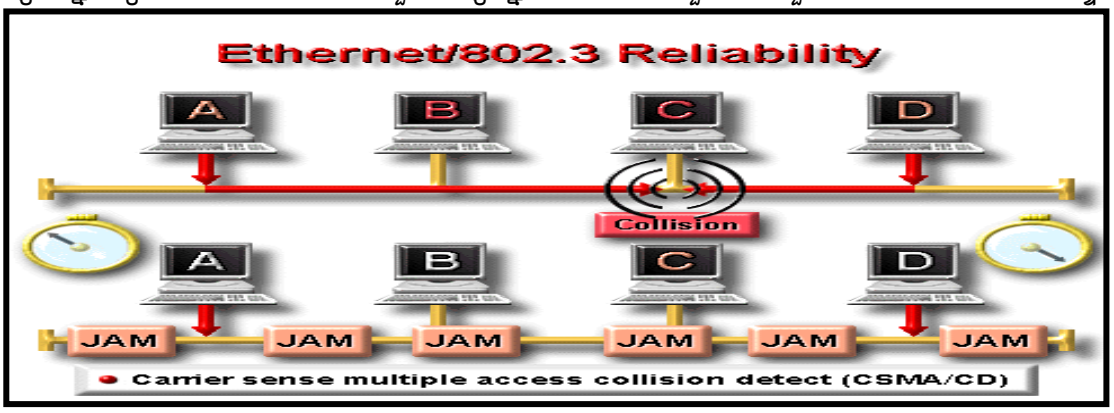
២-៥-ភាពទុកចិត្តរបស់ Ethernet/802.3

CSMA / CD មានតួនាទីដូចខាងក្រោម:

នៅពេលដែលឧបករណ៍មួយមានបំណងចង់ធ្វើការបញ្ជូននោះវាបានត្រួតពិនិត្យបណ្តាញណេតវើកជាមុន សិនដល់ម្បីកំនត់ថាតើឧបករណ៍មួយផ្សេងទៀតកំពុងធ្វើការបញ្ជូនដែរឬទេ។ បើបណ្តាញណេតវើកមិនត្រូវបានប្រាស់នោះទេ ឧបករណ៍ក៏បានចាប់ផ្តើមធ្វើការបញ្ជូន។ ខណៈដែលកំពុងតែធ្វើការបញ្ជូន ឧបករណ៍បានត្រួតពិនិត្យទៅលើបណ្តាញណេតវើកដើម្បីធានាថាគ្មានឧបករណ៍ណាមួយផ្សេងទៀតកំពុងបញ្ជូននោះទេ។ ឧបករណ៍ពីរអាចចាប់ផ្តើមធ្វើការបញ្ជូននៅពេលតែមួយប្រសិនបើឧបករណ៍ទាំងពីរពិនិត្យឃើញថាបណ្តាញណេតវើកនៅទំនេរ។

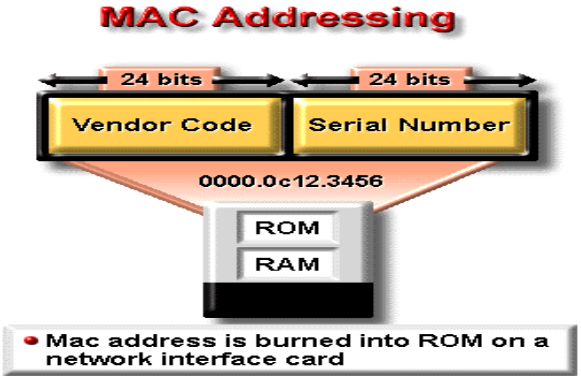
ប្រសិនបើឧបករណ៍ពីរបានធ្វើការបញ្ជូននៅពេលតែមួយនោះការប៉ះទង្គិចគ្នាបានកើតមានឡើងដូចដែលបង្ហាញក្នុងរូបខាងក្រោម។ នៅពេលដែលវាធ្វើការបញ្ជូន node មួយបានទទួលនូវការប៉ះទង្គិច(collision) ហើយនៅពេលនោះ វាបានបញ្ជូននូវសញ្ញាលស្ទះ(Jam signal) មួយដែលបណ្តាលមកពីការប៉ះទង្គិចគ្នាដែលបានប្រើរយៈពេលយ៉ាងយូរសម្រាប់ឲ្យ nodes ផ្សេងៗទៀតដើម្បីទទួលយកវា។

គ្រប់ Nodes ទាំងអស់ដែលបានធ្វើការបញ្ជូន បន្ទាប់មកវាបានបញ្ឈប់ការបញ្ជូន Frame ជាបណ្តោះអាសន្នសិនមុនវាធ្វើការបន្តទៀតឡើងវិញ។ ប្រសិនបើការបញ្ជូនក្រោយទៀតបានផ្តល់លទ្ធផលជាការប៉ះទង្គិច នៅពេលនោះ node បានព្យាយាមធ្វើការបញ្ជូនសារជាថ្មីទៀតចំនួន១៥ដងមុនពេលដែលវាបោះបង់ចោល។ ម៉ោងបង្ហាញពីរយៈពេលនៃការត្រឡប់វិញផ្សេងៗគ្នា។ ប្រសិនបើរយៈពេលមានចំនួនពីរផ្សេងគ្នានោះឧបករណ៍មួយនិងទទួលបានជោគជ័យនៅពេលបន្ទាប់មកទៀត។



MAC Addressing

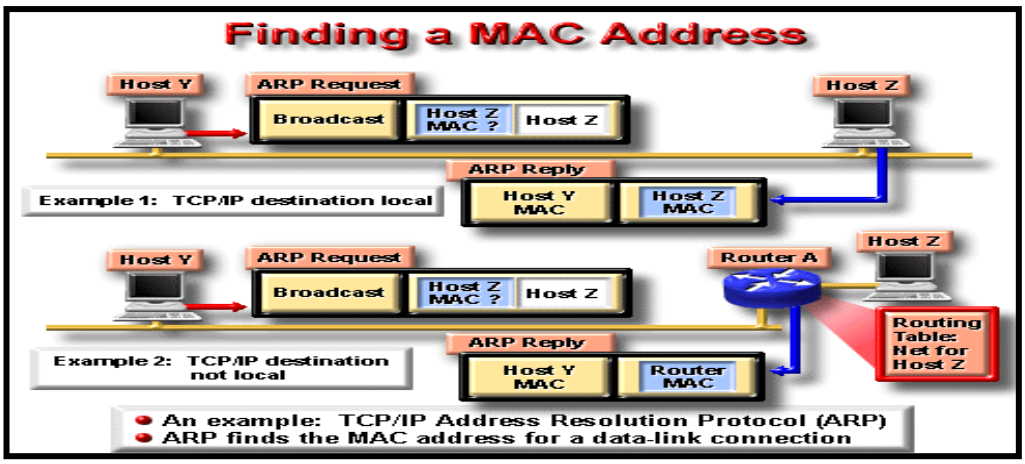
ចំពោះឧបករណ៍ជាច្រើនដើម្បីអាចប្រើប្រាស់នូវខ្សែរួមគ្នាបានហើយព្រមទាំងអាចស្គាល់គ្នាទៅវិញទៅមកបាន។ MAC sublayer បានកំនត់ឲ្យហាដវែរឬ Datalink address មួយដែលត្រូវបានគេហៅថា MAC address។ MAC address មានតែមួយគត់សម្រាប់ LAN interface នីមួយៗ។ MAC address នៃ NIC ត្រូវបានគេបង្កើតឡើងស្ថិតនៅក្នុង ROM ។ នៅពេលដែល NIC បានផ្តល់ឲ្យនូវ Address នេះត្រូវបានចំលងចូលទៅផ្ទុកក្នុង RAM ។



២-៦-ការស្វែងរក MAC address

នៅមុនៅពេលដែលទិន្នន័យជា frame មួយត្រូវបានផ្លាស់ប្តូរជាមួយឧបករណ៍ដោយភ្ជាប់ផ្ទាល់ស្ថិតក្នុង LAN តែមួយ។ ឧបករណ៍បញ្ជូនមាននូវ MAC address មួយដែលវាបានប្រើជាអាស័យដ្ឋានគោលដៅ។ វិធីសាស្ត្រដើម្បីរកឲ្យឃើញនូវ MAC address នៃឧបករណ៍មួយគឺប្រើប្រាស់នូវ ARP(Address Resolution Protocol) ។ តាមរូបភាពបង្ហាញពីវិធីសាស្ត្រពីរនៃ TCP/IP ។ ARP ត្រូវបានប្រើសប្រាស់ដើម្បីរកឲ្យឃើញពី MAC address មួយ។ នៅក្នុងឧទាហរណ៍ទី១ Host Y និង Host Z គឺស្ថិតនៅក្នុង LAN តែមួយ។ Host Y បានបញ្ជូន broadcast នូវ APR ទៅឲ្យ LAN មួយដើម្បីរកឲ្យឃើញនូវ MAC address នៃ HostZ ។ ដោយសារតែ Host Y បានធ្វើការបញ្ជូនចេញនូវ broadcast មួយនោះគ្រប់ឧបករណ៍ទាំងអស់រួមទាំង HostZ និងរកមើលនូវការស្នើសុំនោះ។

ទោះបីជាយ៉ាងណាក៏ដោយមានតែ Hostz តែមួយគត់ដែលបានឆ្លើយតបវិញដោយមានទាំង MAC address របស់វាផងដែរ។ Host Y បានទទួលនូវការឆ្លើយតបរបស់ Host Z ហើយបានរក្សាទុកនូវ MAC address នោះនៅក្នុង Memory ដែលត្រូវបានគេហៅកាត់ថា ARP cahe ។ នៅពេលក្រោយទៀត Host Y ត្រូវបានភ្ជាប់ដោយផ្ទាល់ជាមួយ Host Z ហើយវាបានធ្វើការហៅ MAC addres នៃ Host Z ដែលវាបានរក្សាទុកនោះមកវិញ ។



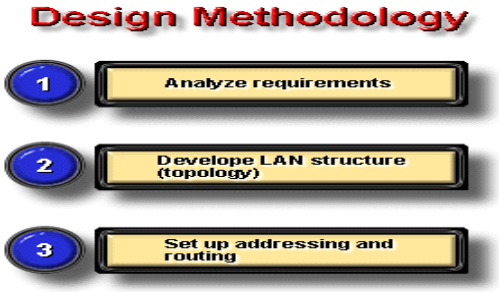
២-៧-គោលបំណងនៃការបង្កើត LAN

ជំហានដំបូងក្នុងការបង្កើត**LAN**មួយគឺគេត្រូវកំនត់ពីគោលបំណងនៃការដំឡើងនោះដោយរៀបចំជាឯកសារ។
ឧទាហរណ៍

- សម្រាប់ចែកចាយនូវ Files និង printers
- សម្រាប់ចែកចាយនូវ Internet
- សម្រាប់បង្កើតជា intranet(Web server និង E-mail server) ដែលជាគោលបំណងទាំងនេះត្រូវបានបំពេញទៅតាមតម្រូវការរបស់អង្គភាពណាមួយឬក្រុមហ៊ុនណាមួយ ប៉ុន្តែតម្រូវការទូទៅសម្រាប់ការបង្កើត**ណេតវើក**មានដូចខាងក្រោម:
  - Functionality: **ណេតវើក**ដែលបានបង្កើតឡើងត្រូចហើយត្រូវតែអាចប្រើប្រាស់បានហើយការប្រើប្រាស់នោះត្រូវតែបំពេញទៅតាមគោលបំណងរបស់អ្នកប្រើប្រាស់ ។
  - Scalability: **ណេតវើក**ដែលយើងបានបង្កើតឡើងរួចហើយត្រូវតែមានពង្រីកបាន ។ វាមានន័យថាបើយើងចង់បន្ថែមនូវចំនួននៃឧបករណ៍ វាមិនអាចធ្វើឲ្យប៉ះពាល់ដល់ការបង្កើតឡើងនោះទេ

- Adatability: **ណេតវើក**ដែលយើងបានបង្កើតឡើងត្រូវតែគិតពីបច្ចេកវិទ្យាថ្មីដែលគេនឹងប្រើប្រាស់នៅពេលខាងមុខ។ មានន័យថា **ណេតវើក**ដែលយើងបានបង្កើតឡើងនោះត្រូវតែអាចប្រើប្រាស់ជាមួយនឹងបច្ចេកវិទ្យាថ្មីបាន
- Manageability: យើងត្រូវតែបង្កើតវាយ៉ាងណាឲ្យមានលក្ខណៈងាយស្រួលក្នុងការគ្រប់គ្រងហើយធានាថាវាអាចដំណើរការបានយ៉ាងល្អ។ ដើម្បីឲ្យមានការគ្រប់គ្រងបានល្អគេត្រូវប្រើ Rack System សម្រាប់រក្សាទុកឧបករណ៍ Networking Devices និងម៉ាស៊ីន Server

មាន៣ដំហានបង្ហាញពីគំរូដែលគេនិយមប្រើក្នុងការដំឡើងប្រព័ន្ធណេតវើក



ជំហានទី១: វិភាគលើតំរូវការ

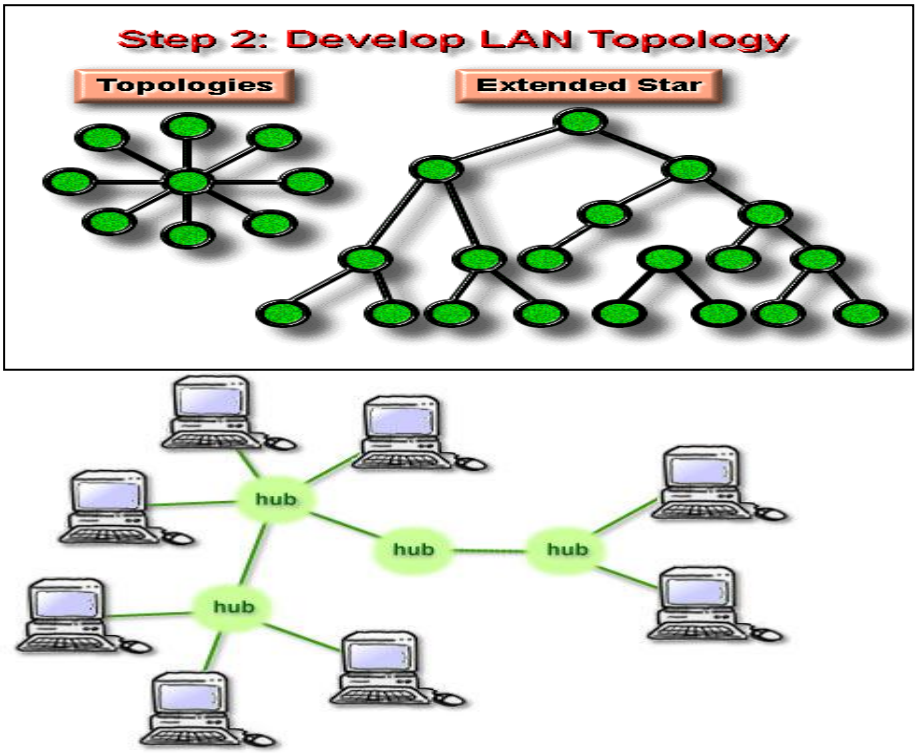
នៅក្នុងវិធីសាស្ត្រនៃការបង្កើតនៃ ណេតវើកគេត្រូវធ្វើការវិភាគទៅលើតម្រូវការនៃ ណេតវើកនិងអ្នកប្រើប្រាស់។ យើងត្រូវគិតទៅលើចំនួននៃអ្នកប្រើប្រាស់ដែលមានការប្រែប្រួល។

ឧទាហរណ៍

ប្រសិនបើប្រព័ន្ធ ណេតវើកមួយបានប្រើប្រាស់នូវ Applications ដើម្បីធ្វើការបញ្ជូននូវសំឡេងនឹងវីដេអូតាមប្រព័ន្ធ ណេតវើក។ ដូច្នេះវាទាមទារឲ្យមាន Bandwidth កម្រិតខ្ពស់។

ជំហានទី២:ការបង្កើតនៃរូបរាងរបស់ LAN

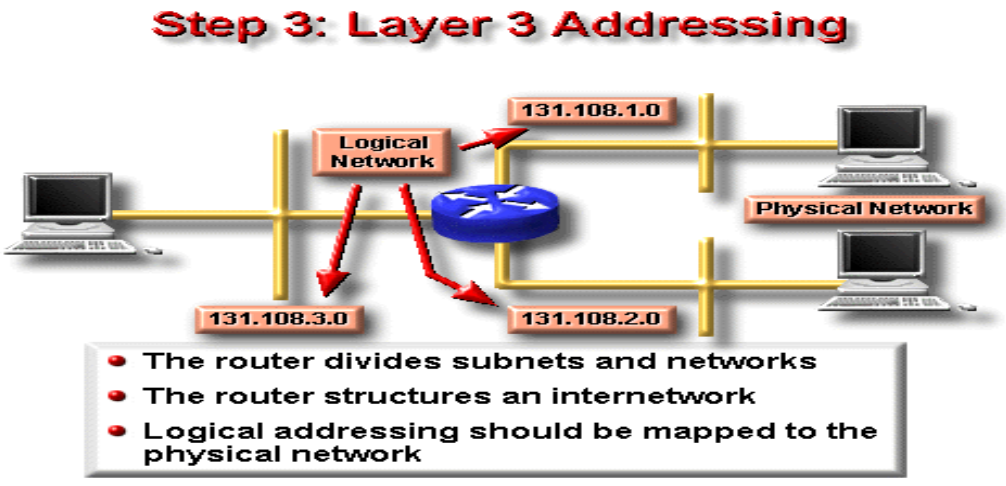
នៅពេលយើងបានដឹងថា Star ឬ Extended start បានប្រើប្រាស់នូវ Ethernet 802.3 CSMA/CD ។ រូបរាងនៃ **ណេតវើក**នេះកំពុងត្រូវបានប្រើប្រាស់យ៉ាងទូលំទូលាយសព្វថ្ងៃនេះ ។



ជំហានទី៣:ការកំនត់ Address នៃ Layer 3

Router អាចឲ្យយើងប្រើប្រាស់នូវបណ្តាញណេតវើកបានកាន់តែទូលំទូលាយពីព្រោះថារ៉ាដើរតួនាទីជា Firewall ដើម្បីទប់ស្កាត់ការ boroadcast ។ ជំហានចុងក្រោយក្នុងការបង្កើត LAN គេត្រូវកំនត់ឲ្យបានច្បាស់លាស់នូវ IP addresses ជាមួយ **ណេតវើក**នេះ ។ គេប្រើ Router ដូចខាងក្រោម:

- ដើម្បីចែក **ណេតវើក**ជា subnet និង network
- សម្រាប់បង្កើតរចនាសម្ព័ន្ធនៃ Internetwork ឬ internetworking
- សម្រាប់ភ្ជាប់ **ណេតវើក**ពីរឬច្រើន ។ ចំពោះរូបរាងនៃ **ណេតវើក**មួយត្រូវការនូវ **ណេតវើក**ដែលជា logical network



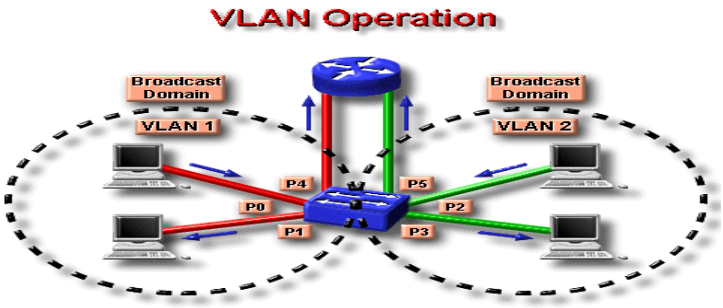
២-៧-១-ការបង្កើត VLAN

ដោយការប្រើប្រាស់នៃVLAN(Virtual LAN) នោះធ្វើឲ្យចរាចរណ៍នៃ broadcast ត្រូវតែមានដែនកំនត់នៅក្នុង VLAN មួយហើយវាបង្កើតបានជាដែនកំនត់នៃ broadcast ឲ្យកានតែតូច។ VLAN ត្រូវបានប្រើប្រាស់ដើម្បីផ្តល់ឲ្យនូវសុវត្ថិភាពដោយបង្កើតបាននូវក្រុមនៃ VLAN ទៅតាមតួនាទីរបស់វា។

ទំហំនៃដែនកំនត់នៃ broadcast មានទំហំកាន់តែតូចហើយការប្រើប្រាស់នៃ Router ដើម្បីទប់ស្កាត់នូវ Broadcast នេះ។ ប្រសិនបើ VLAN1 ចង់បញ្ជូនទៅឲ្យ VLAN2 ដូច្នេះការប្រើប្រាស់វាមានសុវត្ថិភាព។

២-៧-១-២-ដំណើរការរបស់ VLAN

នៅក្នុងឧទាហរណ៍នេះ Port ដែលទាក់ទងត្រូវបានគេប្រើសម្រាប់អនុវត្តនូវការកំនត់របស់ VLAN។ Port P0 P1 និង PC4 ត្រូវបានគេប្រើប្រាស់ដើម្បីផ្តល់ទៅឲ្យ VLAN1។ VLAN 2 មាន Port P2 P3 និង P5។



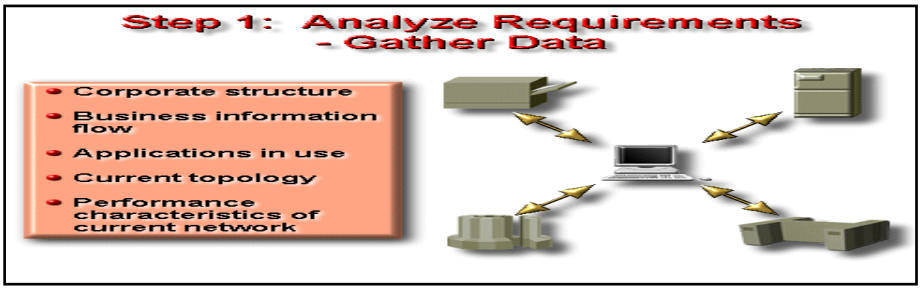
២-៨-ការប្រមូលទិន្នន័យនិងការវិភាគលើតំរូវការរបស់ប្រព័ន្ធ Network

ជំហានដំបូងគឺការប្រមូលព័ត៌មានដែលមានការទាក់ទងទៅនិងរចនាសម្ព័ន្ធនៃសាជីវកម្ម។ នៅដំណាក់កាលចុងក្រោយនៃការបង្កើតត្រូវតែឆ្លើយតបទៅតាមតម្រូវការរបស់រចនាសម្ព័ន្ធនៃសាជីវកម្ម។ ដើម្បីប្រមូលនូវទិន្នន័យដែលយើងត្រូវអនុវត្តដូចខាងក្រោមនេះ។

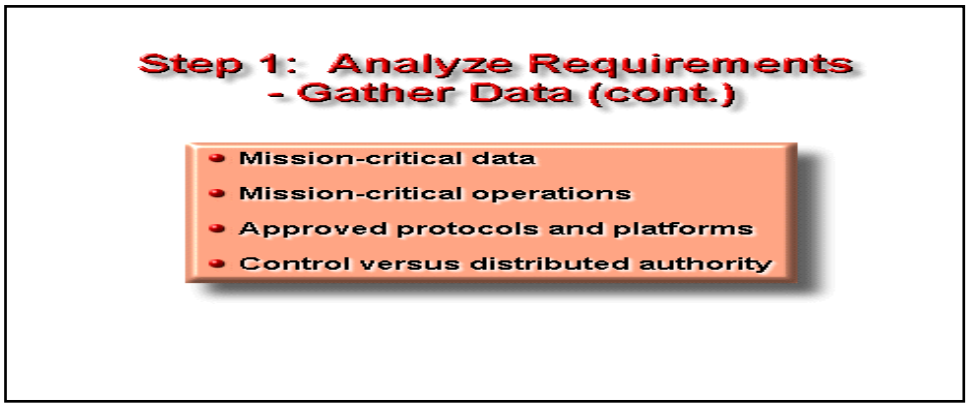
យើងត្រូវពិភាក្សាជាមួយអ្នកប្រើប្រាស់ឲ្យបានច្រើនដងដែលនៅក្នុងនោះយើងត្រូវរកឲ្យឃើញនូវទីតាំងនៃភូមិសាស្ត្រ Applications ដែលគេកំពុងតែប្រើប្រាស់នៅក្នុងគម្រោងដែលមាននៅថ្ងៃខាងមុខ។

ក្រោយពេលដែលយើងបានប្រមូលព័ត៌មានបានរួចរាលហើយដែលទាក់ទងជាមួយនឹងរចនាសម្ព័ន្ធនៃសាជីវកម្ម បន្ទាប់មកយើងត្រូវដឹងពីប្រភពលំហូរព័ត៌មាននៅក្នុងក្រុមហ៊ុន។ យើងត្រូវដឹងពីប្រភពព័ត៌មានដែលបានចែកចាយនិងអ្នកដែលប្រើប្រាស់ទិន្នន័យនោះ។ យើងត្រូវតែស្វែងយល់ឲ្យបានច្បាស់លាស់ថាតើអ្នកប្រើប្រាស់ណាមួយអាចជួយយើងក្នុងការបង្កើត internetwork។ យើងត្រូវតែដឹងពីប្រភពនៃព័ត៌មានដែលបានពីខាងក្រៅ។

ឧទាហរណ៍ការប្រើប្រាស់នូវទិន្នន័យបានមកពី Internet ជាដើម។ យើងត្រូវស្វែងយល់ឲ្យបានស៊ីជម្រៅពីដំណើរការរបស់ប្រព័ន្ធ ណេតវើកណាមួយដែលមានស្រាប់។ បើសិនជាយើងមានពេលគ្រប់គ្រាន់ យើងត្រូវវិភាគទៅលើដំណើរការរបស់ប្រព័ន្ធ interណេតវើកដែលមានស្រាប់។ ផ្នែកដ៏សំខាន់បំផុតក្នុងការបង្កើត interណេតវើកគឺត្រូវតែស្វែងយល់ឲ្យបានច្បាស់លាស់ពីតម្រូវការរបស់អតិថិជន។



យើងត្រូវតែធ្វើយ៉ាងណាឲ្យដឹងថាអ្នកណាក្នុងក្រុមហ៊ុនមានសិទ្ធគ្រប់គ្រងទៅលើ Address ការដាក់ឈ្មោះ ការជ្រើសរើសយករូបរាង ការ Configure លើប្រព័ន្ធជាដើម។ នៅក្នុងក្រុមហ៊ុនខ្លះមានផ្នែកការគ្រប់គ្រងព័ត៌មានវិទ្យា(MIS)ដែលគ្រប់គ្រងទៅលើផ្នែកទាំងអស់នោះ។ មានក្រុមហ៊ុនខ្លះទៀតគេបានធ្វើការចែកនូវMISទៅជាផ្នែកតូចៗជាច្រើនទៀតហើយគេបានផ្តល់ឲ្យនូវសិទ្ធគ្រប់គ្រងទៅតាមផ្នែកនីមួយៗ។



២-៩-វិភាគទៅលើតំរូវការ

ដើម្បីវិភាគទៅលើតម្រូវការរបស់ប្រព័ន្ធ internetwork នោះយើងត្រូវតែវិភាគទៅលើការធ្វើដំនួញរបស់អតិថិជននឹងគោលដៅបច្ចេកទេសជាច្រើនទៀត។

- តើ Applications ថ្មីជាច្រើនដែលបាននិងត្រូវបានគេប្រើយ៉ាដូចម្តេច?
- តើប្រព័ន្ធ**ណេតវើក**ថ្មីជាច្រើននិងត្រូវបានគេចូលប្រើយ៉ាងដូចម្តេច?
- តើដូចម្តេចហៅថាលក្ខខណ្ឌទទួលបានជោគជ័យ?
- តើអ្នកអាចដឹងតាមវិធីណាថាបង្កើតថ្មីទទួលបានជោគជ័យ?

យើងត្រូវវិភាគទៅលើអ្វីត្រូវតែមានជាស្រេចដែលនៅក្នុងនោះមានដូចជា:

-Throughput

- រយៈពេលនៃការឆ្លើយតប

- ការចូលប្រើធនធាន

គ្រប់អតិថិជនទាំងអស់បានឲ្យនិមយន័យនៃតម្រូវការផ្សេងៗគ្នា។ តម្រូវការជាស្រេចត្រូវបានបង្កើតឡើងដោយបន្ថែមនូវធនធានជាច្រើនទៀត។ ធនធានមានតម្លៃកានតែថ្លៃ នោះប្រព័ន្ធ Interណេតវើកបានធ្វើឲ្យការបង្កើតឡើងដើម្បីផ្តល់ឲ្យនូវតម្រូវការជាស្រេចកានតែប្រសើរឡើងហើយចំពោះតម្លៃវិញមិនជាថ្លៃនោះទេ។

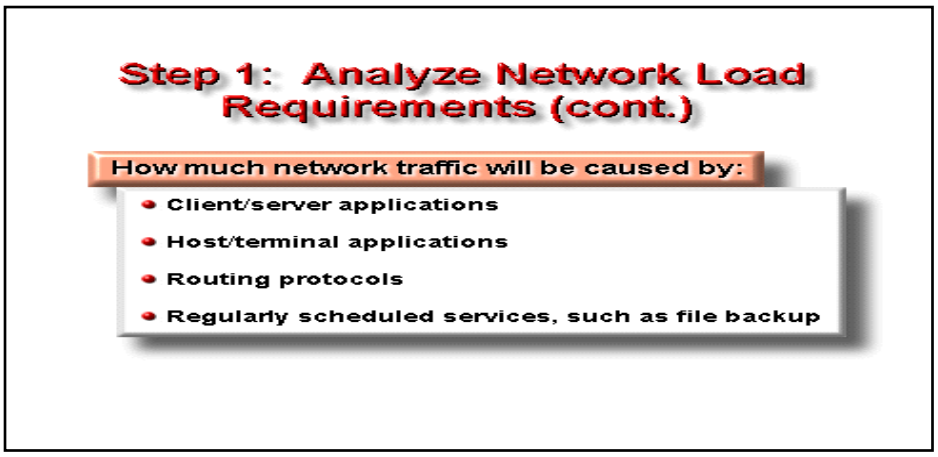
២-៩-១-ការវិភាគទៅលើកត្តាដែលបណ្តាលឲ្យមានចរាចរណ៍ក្នុងប្រព័ន្ធណេតវើក

នៅពេលដែលយើងចាប់ផ្តើមវិភាគទៅលើតម្រូវការផ្នែកបច្ចេកទេសរបស់អតិថិជន យើងត្រូវដឹងអំពីប្រភេទនៃ applications និងប្រភេទនៃ Protocol ដែលបណ្តាលឲ្យមាននូវចរាចរណ៍នៅក្នុងប្រព័ន្ធ**ណេតវើក**។

ឧទាហរណ៍

នៅពេលដែលយើងបានភ្ជាប់នូវ Node មួយទៅក្នុងប្រព័ន្ធ**ណេតវើក**គឺនៅពេលនោះទំហំរបស់ចរាចរណ៍កាន់តែមានទំហំធំជាងមុន។ ដើម្បីបានដឹងពីចរាចរណ៍របស់ប្រព័ន្ធ**ណេតវើក**យើងត្រូវដឹងថានៅពេលណាដែលប្រព័ន្ធ**ណេតវើក**រវល់ជាងគេនិងត្រូវកំនត់នូវសេវាដែលដំណើរការតាមការកំនត់ទុកដូចជា backup file server ជាដើម។

គោលការណ៍:មុនពេលយើងចាប់ផ្តើមបង្កើតប្រព័ន្ធ Inter**ណេតវើក**ហើយព្រមទាំងការប្រើប្រាស់នូវហាដវែ យើងត្រូវតែយល់ឲ្យវបានច្បាស់អំពីចរាចរណ៍របស់ប្រព័ន្ធ **ណេតវើក**។



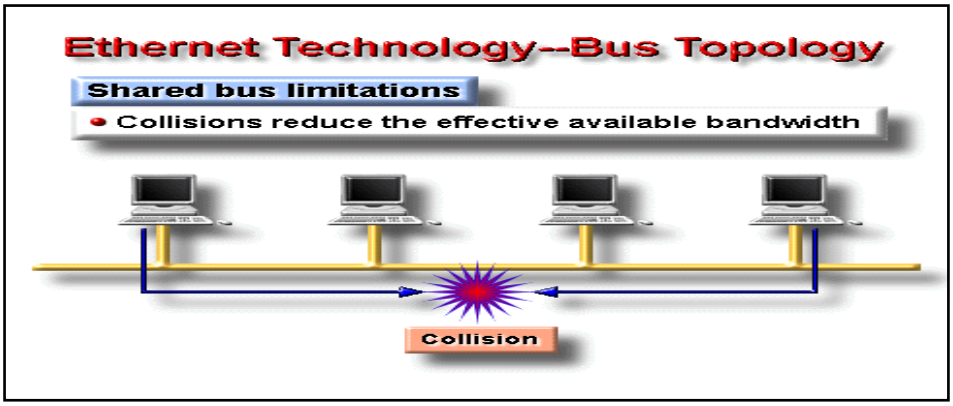
២-៩-២-ការវិភាគទៅលើតំរូវការ-Applications ដែលបណ្តាលឲ្យមានចរាចរណ៍ធំ

ណេតវើកapplications បានបង្កើតឲ្យមានចរាចរណ៍កានតែធំ។ នេះគឺជាតារាងមួយនៃ Aplications ដែលបណ្តាលឲ្យមាននូវការកកស្ទះ (Congestion) ដូចខាងក្រោម:

- ការចូលប្រើ Internet
- កុំព្យូទ័រដែលប្រើ Software ពីចម្ងាយ
- ការបញ្ជូននូវរូបភាពវីដេអូ
- ការប្រើប្រាស់នូវប្រព័ន្ធ Database កណ្តាល
- file serverសម្រាប់ដេប៉ាតឺម៉ង់

គោលការណ៍នៃការដំឡើង:យើងត្រូវដឹងពី Application ដែលបណ្តាលឲ្យមានបញ្ហាចំពោះចរាចរណ៍របស់ ណេតវើក

២-១០-បច្ចេកទេសនៃ Ethernet – ការចែក Network

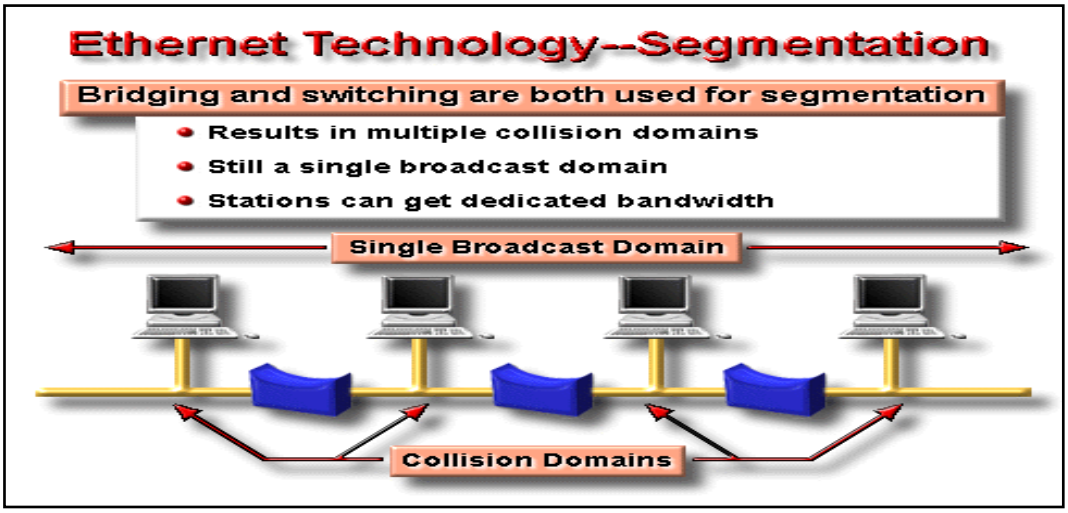


តើមានបញ្ហាអ្វីខ្លះដែលកើតឡើងក្នុងប្រព័ន្ធ Network?

Segmentation គឺជាដំណើរការនៃការចែកដែនកំនត់នៃ Collision មួយទៅជា Collision domain ពីរឬច្រើន។

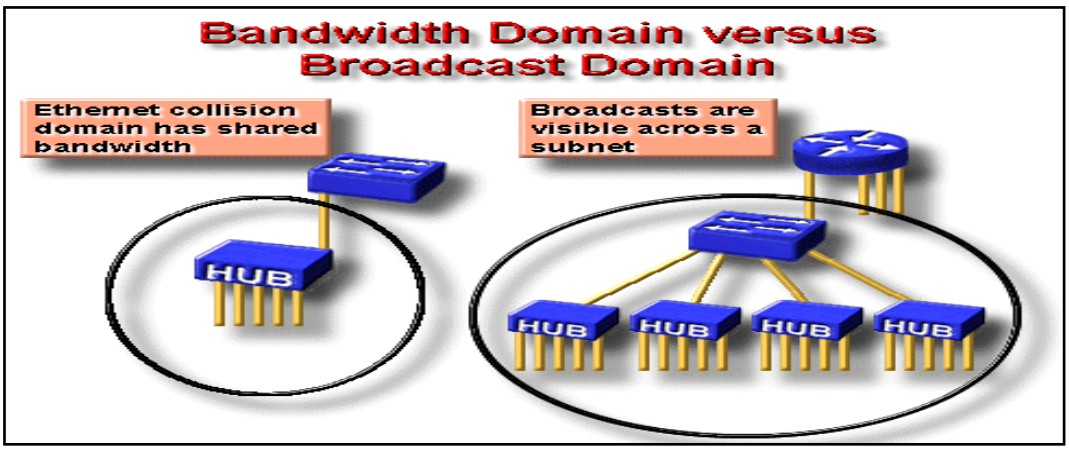
- Layer 2 bridging ត្រូវបានគេប្រើប្រាស់សម្រាប់ចែកនូវរូបរាងនៃ BUS ហើយព្រមទាំងបានបំបែកនូវ Collision domain ជាច្រើននិងបានផ្តល់លទ្ធផលឲ្យឧបករណ៍នីមួយៗមាននូវ Bandwith គ្រប់គ្រាន់សម្រាប់ប្រើប្រាស់បាន

☞ ចំណាំ: នៅក្នុងរូបរាងជា BUS ទាំងមូលមាននូវដែនកំនត់នៃ Broadcast តែមួយគត់ពីព្រោះថា ទោះបីជាយើងប្រើប្រាស់នូវ Bridge និង Switch ក៏ដោយ។



២-១0-Bandwidth Domain ប្រៀបធៀបជាមួយ Broadcast Domain

ដែនកំនត់នៃ Bandwith មួយត្រូវបានគេភ្ជាប់ទៅនិង Port មួនៃ Bridge មួយឬ Switch មួយ។ នៅក្នុងករណីវាជា Ethernet Switch មួយនោះដែនកំនត់នៃ Bandwidth មួយត្រូវបានគេហៅថា Collision domain ផងដែរ។ ដែនកំនត់នៃ Broadcast មួយត្រូវតែភ្ជាប់ជាមួយ Port មួយនៃ Router មួយ។ គ្រប់ឧបករណ៍ទាំងអស់នៅក្នុងដែនកំនត់នៃ Bandwidth មួយបានប្រើប្រាស់រួមគ្នានៅក្នុងធនធានរបស់ LAN bandwidth តែមួយ។ គ្រប់ចរាចរណ៍ទាំងអស់បានមកពី Host ណាមួយនៅក្នុងដែនកំនត់នៃ Bandwidth វាជាការមើលឃើញទៅគ្រប់ Hosts ផ្សេងៗទៀត។ នៅក្នុងករណីនៃដែនកំនត់របស់ Ethernet Collision មួយនោះឧបករណ៍ចំនួនពីរអាចធ្វើការបញ្ជូននៅពេលតែមួយដែលបណ្តាលឲ្យមាន Collision កើតមានឡើង។ គ្រប់ broadcast ទាំងអស់បានមកពី Host ណាមួយនៅក្នុង broadcast domain តែមួយវាជាការមើលឃើញទៅគ្រប់ Hosts ផ្សេងៗទៀតនៅក្នុងដែនកំនត់នៃ Broadcast តែមួយ។ Broadcasts ត្រូវតែមើលឃើញចំពោះគ្រប់ Hosts ទាំងអស់នៅក្នុងដែនកំនត់នៃ broadcast តាមលំដាប់លំដោយដើម្បីបង្កើតបានជាសកម្មភាពនៃការភ្ជាប់។



២-១១-គោលការណ៍នៃការដំឡើងប្រព័ន្ធ Network

២-១១-១-ការបង្កើតនូវ LAN Topology

ក្រោយពីយើងបានប្រមូលនូវព័ត៌មានដែលទាក់ទងទៅនិងប្រព័ន្ធ Networkបានរួចរាល់ហើយ នោះយើងត្រូវជ្រើសរើសប្រភេទនៃរូបរាងឬគំរូនៃ LAN មួយដែលត្រូវបានបង្កើតឡើងនោះ។ ផ្នែកដ៏សំខាន់នៃការបង្កើតនៃរូបរាងនេះត្រូវបានបំបែកទៅជា៣ប្រភេទនៃ OSI Model ។

- Layer 1- Physical Layer

គោលបំណងនៃការបង្កើត:ការបង្កើតនូវ Layer នេះនៃ OSI Model ដែលមានល្បឿនលឿននឹងការបង្កើតនូវសមត្ថភាពដែលអាចប្រើប្រាស់បាន។

- Layer 2- Data Link Layer

នៅក្នុង Layer នេះឧបករណ៍ដូចជា Bridge ឬ LAN Switch ត្រូវបានគេប្រើសម្រាប់ការភ្ជាប់ LAN segment តាមរយៈនៃខ្សែ។ ឧបករណ៍ដែលដំណើរការនៅក្នុង Layer នេះមានតួនាទីសម្រាប់កំនត់នូវទំហំនៃ Collision និងដែនកំនត់នៃ Broadcast ។

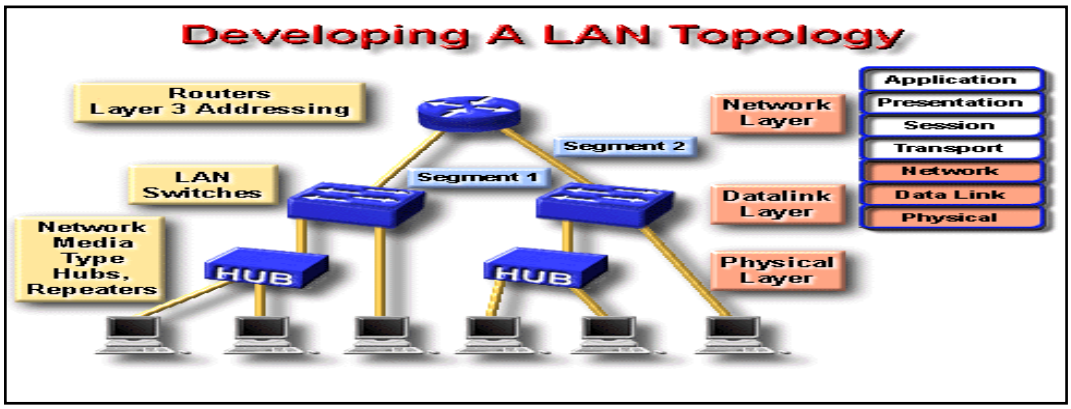
គោលបំណងនៃការបង្កើត: សម្រាប់បង្កើតនូវចំណុចកណ្តាកមួយដែលមាននៅក្នុងនោះរួមាន MDFឬ IDF ដែល Host ត្រូវបានបង្កើតជាក្រុមតាមរយៈនៃ Physical Layer ដើម្បីបានទម្រង់ជា Physical LAN segment មួយ។ ការដំឡើងនូវឧបករណ៍នៃ LAN Switch ដែលបានប្រើប្រាស់នូវផ្នែកតូចៗដើម្បីកាត់បន្ថយទំហំនៃដែនកំនត់របស់ Collision ដែលកើតមានឡើង។ ការបង្កើតឲ្យមាននូវចំណុចមួយនៅ Layer 2 នៃរូបរាងនេះដែលអ្នកប្រើប្រាស់ត្រូវបង្កើតឡើងជាក្រុមទៅជា VLAN(Virtual LAN) ហើយមាននូវដែនកំនត់នៃ Broadcast តែមួយគត់ ។

- Layer 3- network Layer

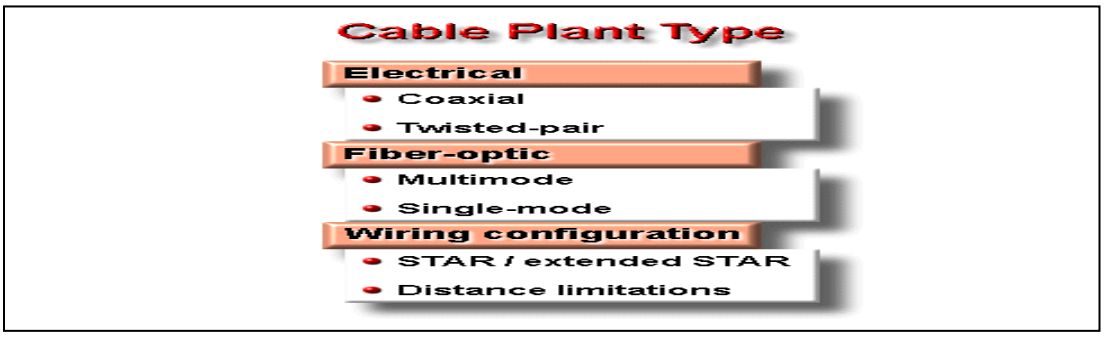
នៅក្នុង Layer នេះឧបករណ៍ដូចជា Router ត្រូវបានប្រើប្រាស់ដើម្បីបង្កើតបានជាLAN segment មួយហើយអនុញ្ញាតឲ្យមានការភ្ជាប់រវាង Segment នៅ Layer 3 ដូចជា IP address ជាដើម។

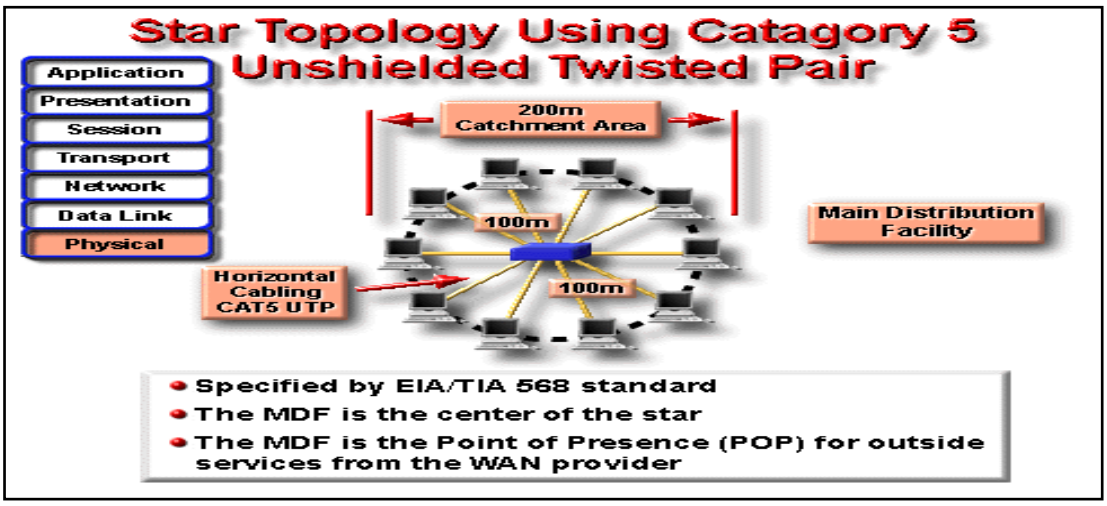
គោលបំណងនៃការបង្កើត: បង្កើតនូវផ្លូវមួយរវាង LAN segment ដែលបានប្រាៈនូវ Data packet ។

- ចែក ARP Protocol Broadcast ជាច់ស្រឡះពីគ្នា
- ចែកឲ្យជាច់ស្រឡះពីគ្នានៃ Collision រវាង Segments.
- ប្រាៈសេវានៃ layer ទី 4 រវាង Segments



ប្រភេទនៃខ្សែរបស់Network





๒-๑๑-๒-Extended Star Topology

នៅក្នុងប្រព័ន្ធណេតវើកមួយដែលកាន់តែធំជាធម្មតាវាត្រូវការនូវបន្ទប់សម្រាប់រក្សាទុកខ្សែច្រើនជាងមួយ ប្រសិនជាមានចំនួន Hosts កាន់តែច្រើនឡើងដែលត្រូវការភ្ជាប់ជាណេតវើក បុន្តែការភ្ជាប់ទៅខាងក្រៅមានចម្ងាយត្រឹមតែ១០០ម៉ែត្រប៉ុណ្ណោះចំពោះ CAT5 UTP Ethernet ។

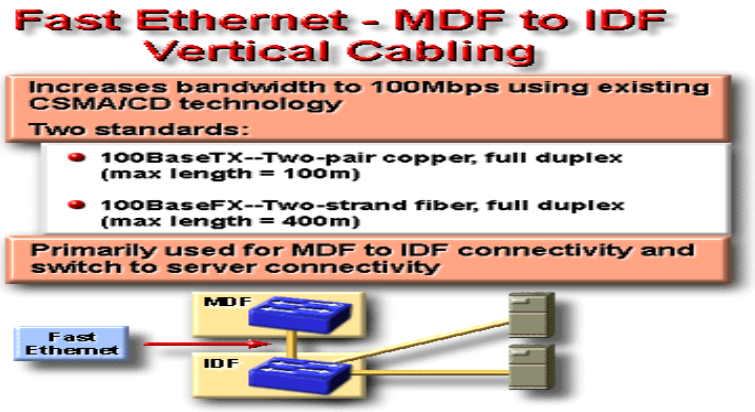
ដោយត្រូវការបន្ថែមនូវបន្ទប់ខ្សែ(Wiring Closet) នោះតំបន់សម្រាប់ភ្ជាប់ជាច្រើនត្រូវបានបង្កើតឡើង។ បន្ទប់ខ្សែទី២ត្រូវបានផ្តល់នូវ IDF(Intermedate Distrubution Facilities) ។ ស្តង់ដា EIA/TIA 568 បានបង្កើតឡើងនោះមាន IDFs ត្រូវបានភ្ជាប់ទៅ MDF ដោយប្រើប្រាស់នូវVC(Vertical Cabling) ។ VC នេះគឺជាប្រភេទនៃ Fiber Optics ពីព្រោះថាដោយសារតែ Fiber Optic អាចប្រើប្រាស់បានចម្ងាយកាន់តែឆ្ងាយ ។ នៅក្នុង MDFs និង IDFs ផ្នែក

ដ៏សំខាន់ផ្សេងៗគ្នាមានការប្រើប្រាស់ Patch Panel មួយផ្សេងទៀតដែលជា VCC(Vertical Cross Connect)។ VCC នេះត្រូវបានប្រើប្រាស់ដើម្បីភ្ជាប់ IDFs ជាច្រើនទៅ MDF កណ្តាល។ ប្រវែងនៃ VC គឺជាប្រភេទនៃខ្សែដែលមានប្រវែងវែងជាង ១០០ម៉ែត្រចំពោះ CAT 5 UTP ចំពោះ Fiber Optic វិញត្រូវបានប្រើប្រាស់ជាធម្មតា។

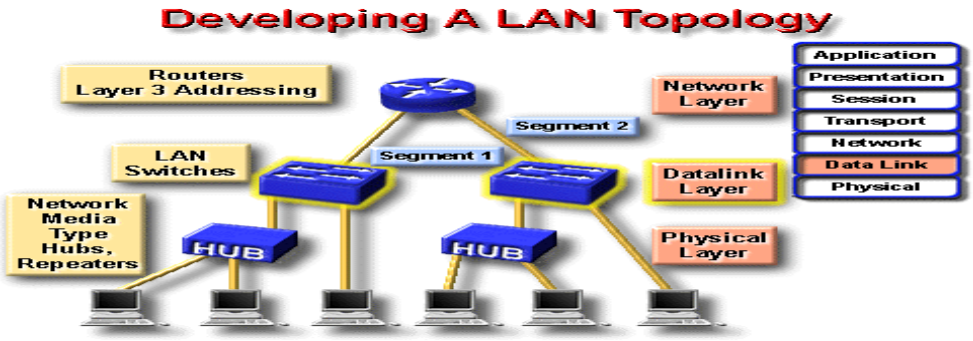
២-១២-ការភ្ជាប់ពីFast Ethernet-MDF to IDF

Fast Ether គឺជា Ethernet ដែលត្រូវបានបង្កើតឡើងដែលមានល្បឿនរហូតដល់ 100 Mbps។ ប្រភេទនេះបានប្រើប្រាស់នូវស្តង់ដា Ethernet ដែលមានរាងជា BUS ដែលមាន Broadcast ដែលយើងប្រើនូវ 10 base T ។

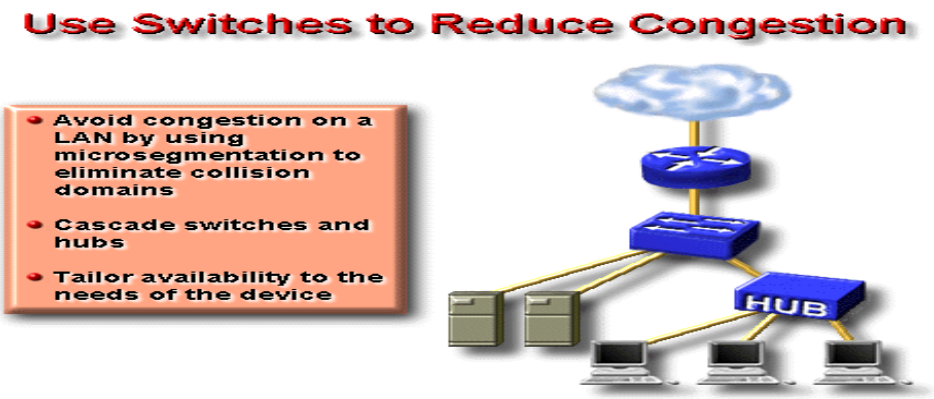
CSMA/CD Method គឺជា Media Access Control (MAC) មួយ។ ស្តង់ដា Fast Ethernet មាន៨ផ្សេងៗគ្នាដូចជា Copper-Pair Wire (100Base TX) និង Fiber (100Base FX) ហើយគ្មាន Fast Ethernet ដែលប្រើខ្សែ Coaxial នោះទេ។



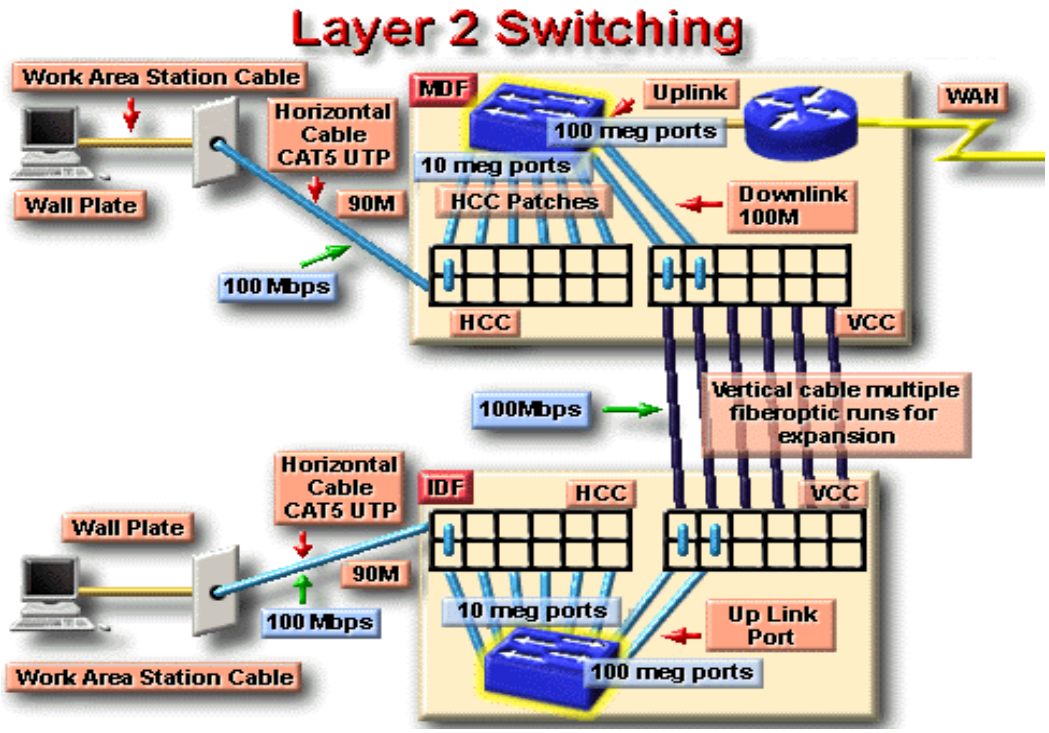
LAN Swith ដែលមានពីរស្រទាប់នៃ OSI Model



២-១២-១-ការប្រើប្រាស់ Switchs ដើម្បីកាត់បន្ថយការកកស្ទះ(Congestion)

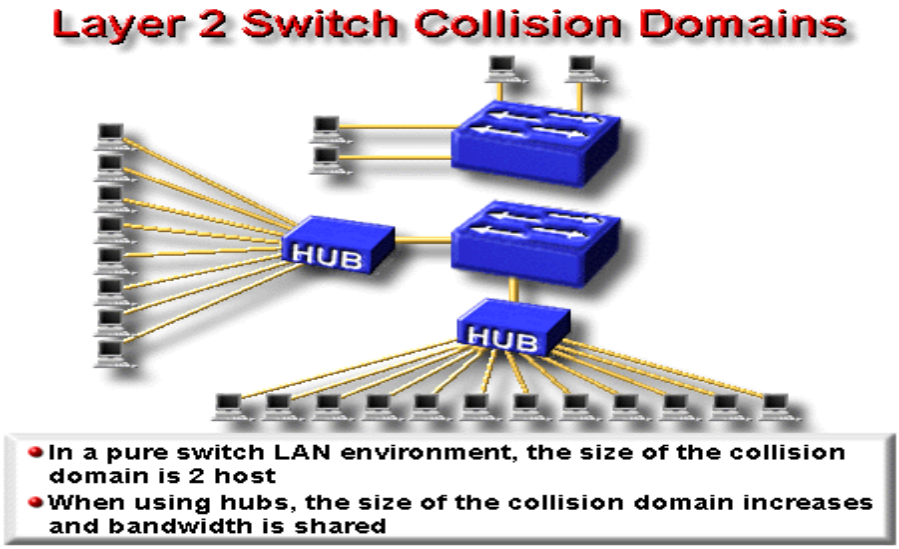


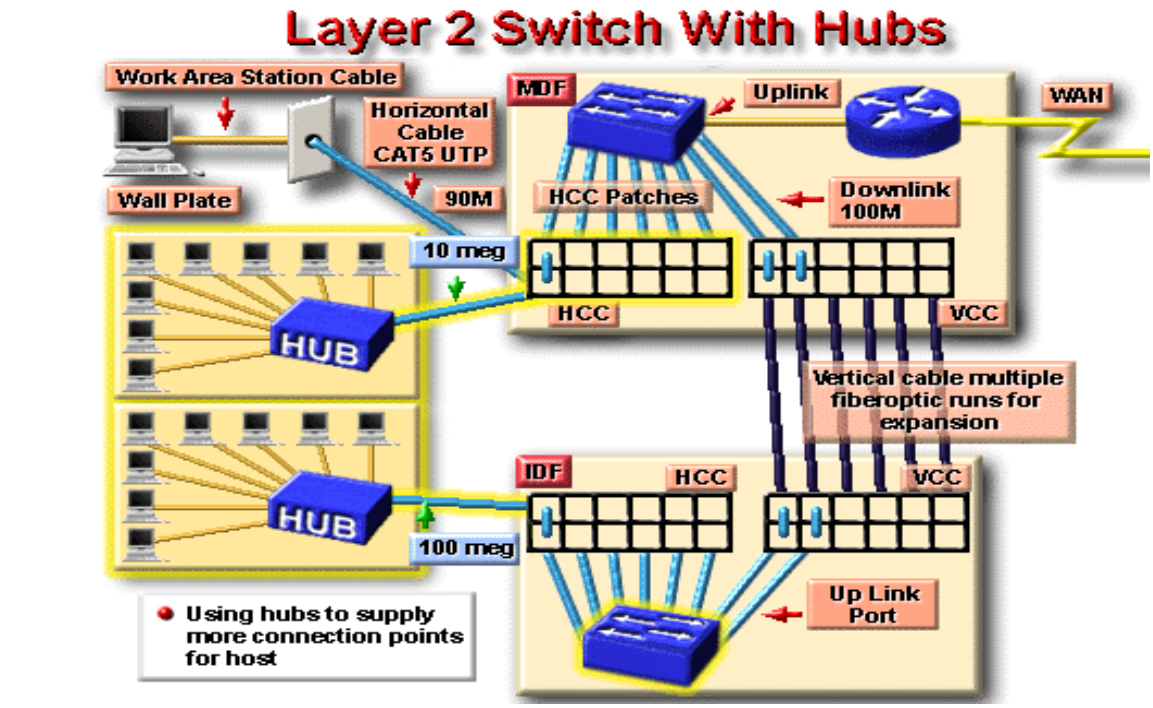
Microsegmentation មានន័យថាការប្រើប្រាស់នូវ Bridge និង Switch ដើម្បីឲ្យដំណើរការរបស់ Workgroup ឬ Backbone ណេតវើកបានល្អប្រសើ។ បើ Switch ត្រូវបានប្រើប្រាស់ជាមួយ Hub នោះដើម្បីឲ្យដំណើរការរបស់ប្រព័ន្ធណេតវើកបានល្អប្រសើរសម្រាប់អ្នកប្រើប្រាស់ចាំបាច់និង Server ។



២-១២-២-Layer 2 Switch Collision Domains

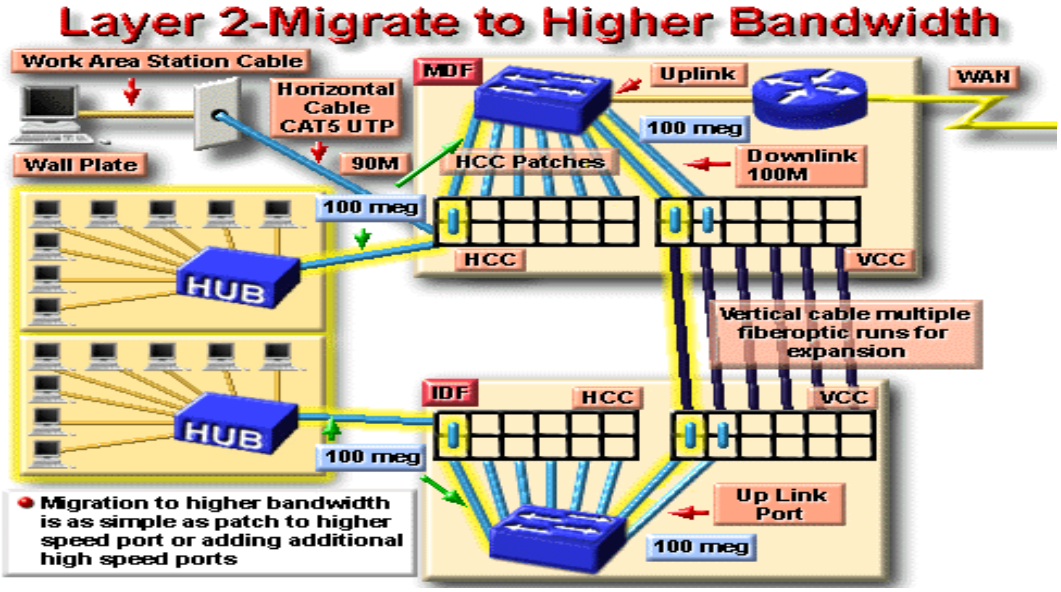
ដើម្បីកំនត់ពីទំហំនៃដែនកំនត់របស់ Collision មួយយើងត្រូវដឹងពីចំនួន Hosts ដែលត្រូវភ្ជាប់ទៅ Port តែមួយរបស់ Switch។ ការកំនត់នេះវាមានការទាក់ទងជាមួយនិង Bandwidth ដែលបម្រុងទុកសម្រាប់ Host នីមួយៗ។ នៅក្នុងស្ថានភាពបែបនេះមាន Host តែមួយគត់ដែលបានភ្ជាប់ទៅ LAN switch port។ យើងត្រូវតែកំនត់នូវទំហំនៃដែនកំនត់របស់ Collision ចំនួនពីរ(Source Host និង Destination Host)។ ដោយសារតែទំហំនៃដែនកំនត់របស់ Collision កាន់តែតូចនោះវាគ្មាន Collision កើតឡើងនោះទេនៅពេលដែល Host ពីរបានកំពុងតែបញ្ជូនឲ្យគ្នាទៅវិញទៅមក។





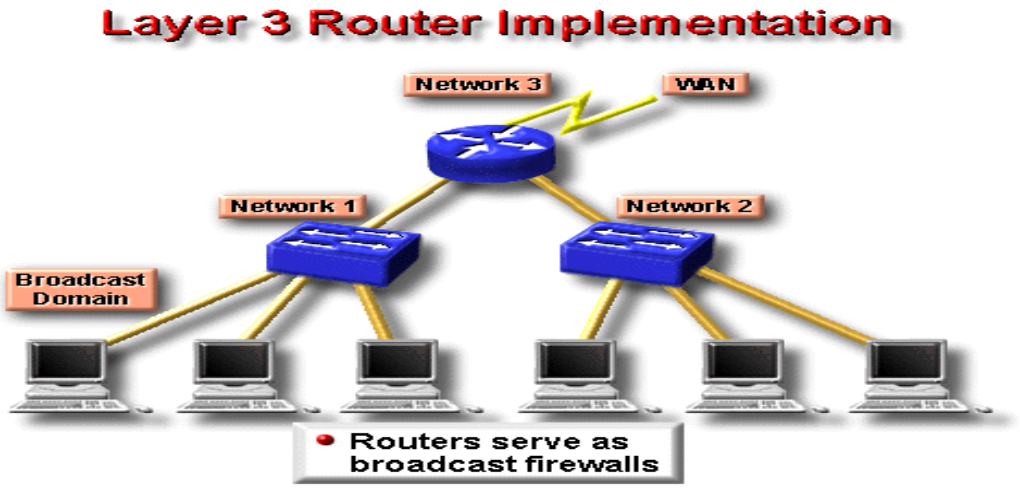
២-១២-៣-Layer 2 Migration ដើម្បីឲ្យបាន Bandwidth ខ្ពស់

Migration គឺជាការបញ្ចូលគ្នានៃការប្រើប្រាស់ដើម្បីឲ្យការប្រើប្រាស់បានល្អប្រសើរ។ យើងត្រូវការភ្ជាប់ Host មួយទៅPortមួយរបស់ Switch។ នៅពេលដែលប្រព័ន្ធណេតវើកមួយកាន់តែធំនោះតម្រូវការនៃ Bandwidth ក៏កាន់តែធំដែរ។ VCC ត្រូវបានគេប្រើប្រាស់ដើម្បីភ្ជាប់ MDF ជាមួយ IDF។ ដូច្នេះគេត្រូវការនូវ bandwidth សម្រាប់ Port នោះចំនួន 100Mbps ហើយនៅពេលដែលយើងបានប្រើប្រាស់នូវ Switch ដែលមានពីរស្រទាប់នោះវាបានផ្តល់ឲ្យយើងនូវ Bandwidth គ្រប់គ្រាន់សម្រាប់ Migration។ នៅក្នុងខ្សែ HC(Horizontal Cabling)bandwidth បានកើនឡើងហើយការផ្លាស់ប្តូរពី HCC ទៅជា Port មួយនៃ Switch ដែលមានល្បឿន 100 Mbps។ ការកំនត់នូវទំហំនៃ layer 2 LAN switch វាមានសារៈសំខាន់ណាស់ដើម្បីឲ្យមាន bandwidth ដល់ 100 Mbps port សម្រាប់ Migration ដែលត្រូវតែការនូវ bandwidth កាន់តែធំ។



២-១២-៤-Layer 3 Routing Implementation

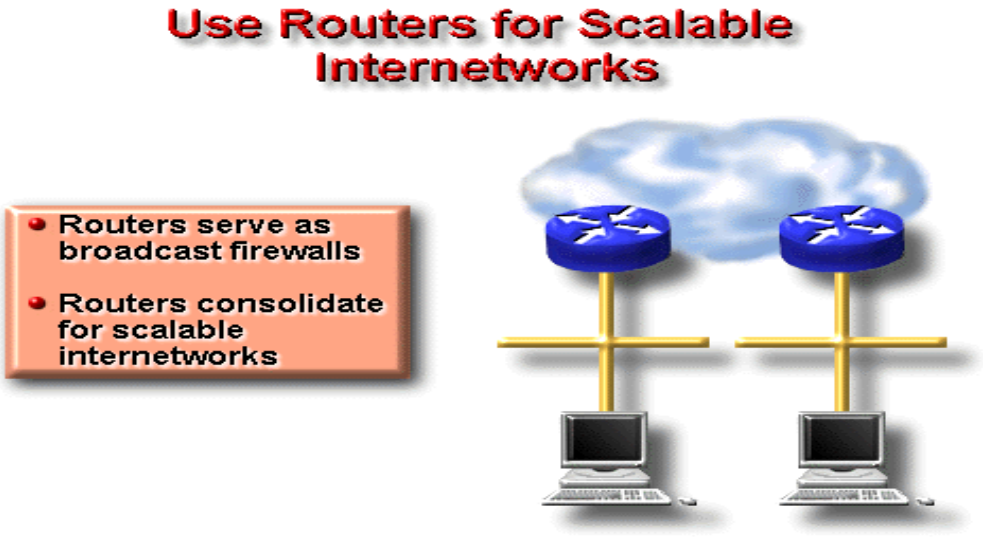
ការប្រើប្រាស់ឧបករណ៍ដែលមាន៣ស្រទាប់ដូចជា Router បានអនុញ្ញាតឲ្យមានការបំបែកជាផ្នែកនៃ LAN នៅក្នុងប្រព័ន្ធណេតវើករួមមានទាំងផ្នែករូបរាងនិងផ្នែកខាងក្នុង។ Routers ត្រូវបានប្រើប្រាស់សម្រាប់ភ្ជាប់ទៅជាមួយ WAN(Wide Area Network) ដូចជា Internet ជាដើម។



២-១២-៥-ការប្រើប្រាស់ Router ដើម្បីឲ្យ Internetworks កាន់តែធំ

បច្ចេកវិទ្យា Routing បានច្រោះនូវការ Broadcast នៃ Data Link និង Multicast ។ ដោយសារតែមានការបន្ថែមនូវ Port របស់ Router ដែលមានការបន្ថែមនូវ Subnet ឬ network addressនោះប្រព័ន្ធ Internetwork ត្រូវបានចែកចេញជាផ្នែកៗទៅតាមតម្រូវការ។ ណេតវើក Protocol Addressing និង Routing បានផ្តល់នូវការបង្កើតប្រព័ន្ធមួយដែលអាចពង្រីកបាន។ នៅពេលធ្វើការសម្រេចចិត្តថាតើត្រូវប្រើប្រាស់នូវ Router ឬ Switch នោះ យើងត្រូវដឹងតើមានអ្វីកើតមានឡើងនៅពេលដែលយើងកំពុងតែព្យាយាមដោះស្រាយបញ្ហានោះ?

បើសិនជាបញ្ហារបស់អ្នកជាបញ្ហានៃ Protocol វិញនោះ ទាក់ទងទៅនិងបញ្ហាកកស្ទះ។ Router ត្រូវបានប្រើប្រាស់សម្រាប់ដោះស្រាយនូវរាល់បញ្ហាដែលមានបរិមាណនៃចំនួន Broadcast ច្រើនហើយនិង Protocol ដែលមិនអាចពង្រីកបាន បញ្ហាផ្នែកសុវត្ថិភាព បញ្ហាផ្នែកការកំនត់ IP address ។ ទោះបីជាយ៉ាងណាក៏ដោយ Router មានតម្លៃថ្លៃហើយមានលំបាកក្នុងការ Configure ជាង Switches ។



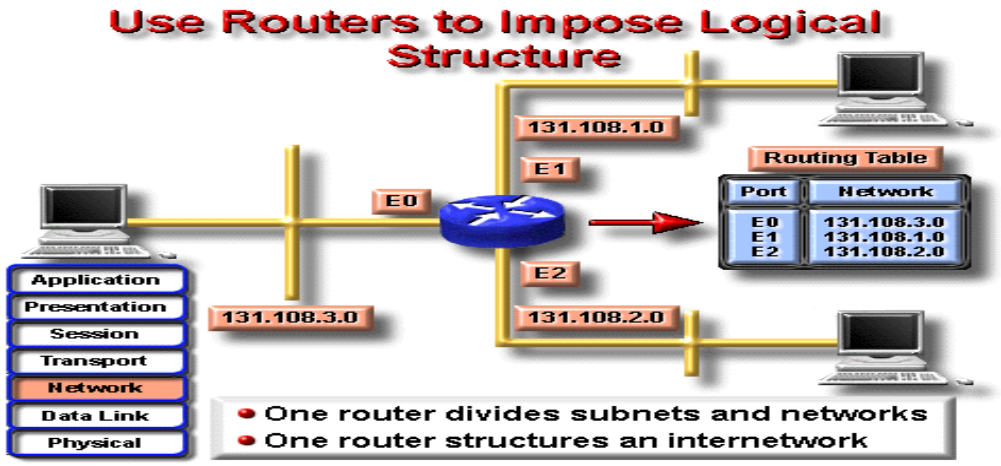
២-១២-៦-ការប្រើប្រាស់ Router ដើម្បីឲ្យរចនាសម្ព័ន្ធផ្នែកខាងក្នុងបានល្អប្រសើរ

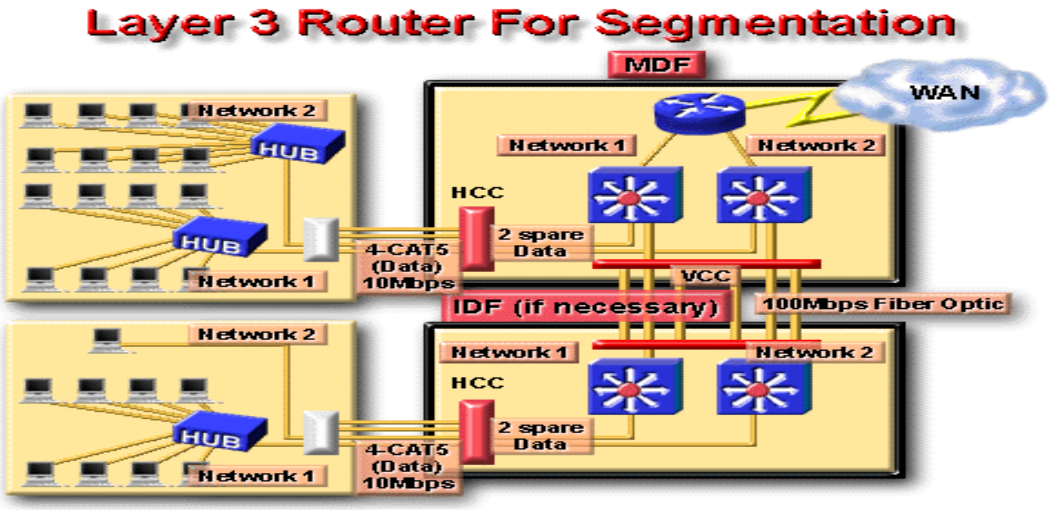
Router ត្រូវបានប្រើប្រាស់ដើម្បីធ្វើឲ្យមានលទ្ធភាពក្នុងការពង្រីកឲ្យកាន់តែធំបានល្អពីព្រោះវាត្រូវបានប្រើ Firewall សម្រាប់ការពារកុំឲ្យមាននូវ Broadcast ។ លើសពីនេះទៅទៀត Router បានធ្វើឲ្យមាននូវសមត្ថភាពនៃការពង្រីកបានកាន់តែប្រសើរដោយសារតែ Layer 3 ។

ឧទាហរណ៍

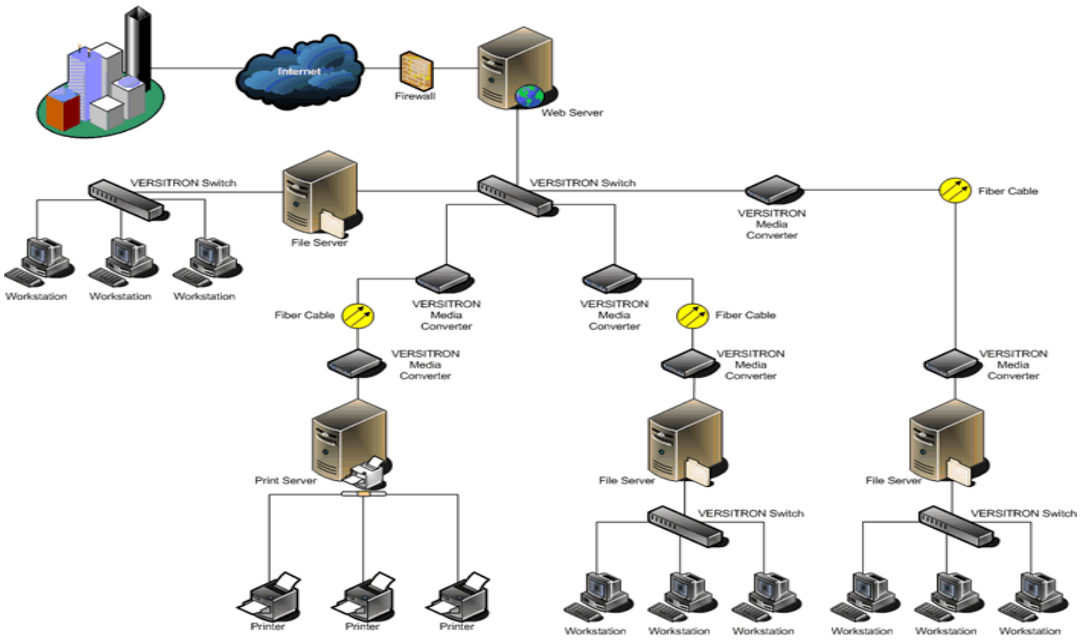
IP subnet បានបន្ថែមនូវរចនាសម្ព័ន្ធទៅឲ្យ Addresses។ ដោយសារតែ Bridge និង Switches មិនបានស្គាល់នូវAddress បាននោះទេ។ ដូច្នេះវាបានបញ្ជូនចេញទៅគ្រប់ Port ទាំងអស់។

ជាមួយនិង Routers Hosts បានប្រើនូវ Protocol ដែលបាននូវ network layer អាចដោះស្រាយបញ្ហាក្នុងការរក Address នៃ Host ផ្សេងៗទៀតដោយមិនបញ្ជូនទៅឲ្យជា Broadcast នោះទេ។ បើសិនជាអាស័យដ្ឋានរបស់គោលដៅជា IP address វិញនោះការបញ្ជូន Host អាច Encasulate ចំពោះ Packet នៅក្នុង Data Link header មួយនិងបានបញ្ជូន Unicast Frame ដោយផ្ទាល់ទៅឲ្យឧបករណ៍មួយ។





២-១២-៧-ការ Design ណេតវើកដោយប្រើ Fiber Optic

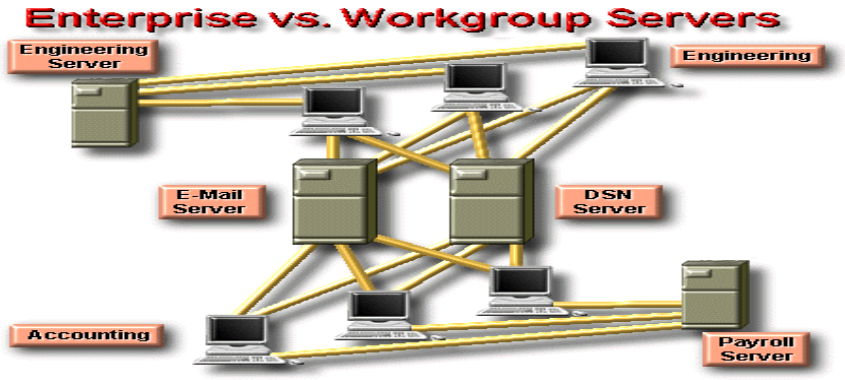


File Servers and ចារចរណ៍ Patterns

២-១២-៧-១-Enterprise ប្រៀបធៀបជាមួយ Workgroup Servers

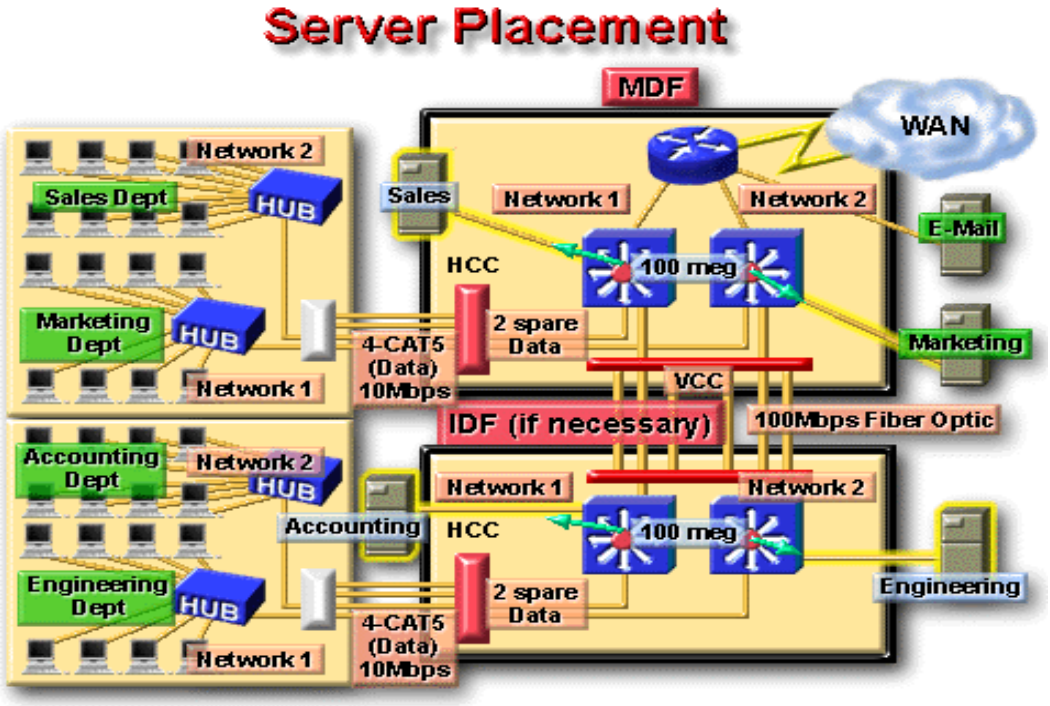
ដើម្បីធ្វើការដំឡើងប្រព័ន្ធ ណេតវើកមួយឲ្យទទួលបានជោគជ័យនោះ យើងត្រូវតែយល់ឲ្យបានច្បាស់ពីចរាចរណ៍ទិន្នន័យនៃប្រព័ន្ធ Network ។ Application server ត្រូវបានចែកចេញជាពីរប្រភេទគឺ Enterprise Server និង Workgroup server ។ Enterprise Server ប្រើប្រាស់សម្រាប់អ្នកប្រើប្រាស់ទាំងអស់នៅក្នុងប្រព័ន្ធ ណេតវើកទាំងមូលដូចជា E-mail server ឬ DNS server ។ ចំណែកឯ Workgroup server វិញត្រូវបានប្រើប្រាស់សម្រាប់ក្រុមនៃអ្នកប្រើប្រាស់ផ្នែកណាមួយនៃ Nerwork ។

អ្នកប្រើប្រាស់ទាំងអស់នៅក្នុងប្រព័ន្ធណេតវើកត្រូវការចូលប្រើចំពោះ Enterprise Server វាត្រូវតែភ្ជាប់ទៅនិង MDF ។ ចរាចរណ៍ចំពោះប្រភេទនៃសេវានេះមានតែមួយគត់ដែលត្រូវតែបញ្ជូនទៅកាន់ MDF ហើយវាមិនត្រូវបានបញ្ជូនឆ្លងកាត់ណេតវើកផ្សេងៗទៀតនោះទេ ។ Workgroup server ត្រូវមានទីតាំងនៅក្នុង IDF ។ ចំពោះអ្នកប្រើប្រាស់ទាំងអស់ដើម្បីអាចចូលប្រើនូវ Application ទាំងនោះបាន ។ ដោយការធ្វើការបែបនេះ ចរាចរណ៍ចំពោះ Servers ទាំងអស់នេះត្រូវតែមានមួយគត់ត្រូវបានបញ្ជូនទៅកាន់ណេតវើកទៅកាន់ IDF និងមិនធ្វើឲ្យមានការប៉ះពាល់ដល់អ្នកប្រើប្រាស់ផ្សេងៗទៀតនៅក្នុងប្រព័ន្ធផ្នែកនៃ Segment ទេ ។



២-១២-៧-២-ការកំណត់ទីតាំងនៃ Server

ប្រសិនបើ Server ត្រូវបានផ្តល់ឲ្យនូវណេតវើកហើយវាអាស្រ័យទៅតាមតួនាទីរបស់វា។ network layer 2 និង៣ត្រូវបានបង្កើតឡើងដើម្បីផ្តល់ឲ្យតាមតម្រូវ។ នៅក្នុង MDFs និង IDFs គឺ Layer 2 LAN Switch ត្រូវតែមានល្បឿនខ្ពស់គឺ 100Mbps port ដែលត្រូវប្រើសម្រាប់ Server ទាំងអស់នោះ។



សំណួរត្រួតពិនិត្យ

១-តើ MDF ជាអ្វី?

២-តើ IDF គឺជាអ្វី?

៣-តើ Server ចែកចេញជាប៉ុន្មានប្រភេទ?អ្វីខ្លះ?

៤-Broadcast domain និងBandwidth domain គឺជាអ្វី?

៥-តើ Congestion គឺជាអ្វី? ហើយដើម្បីកាត់បន្ថយ Congestion តើគេត្រូវធ្វើដូចម្តេច?

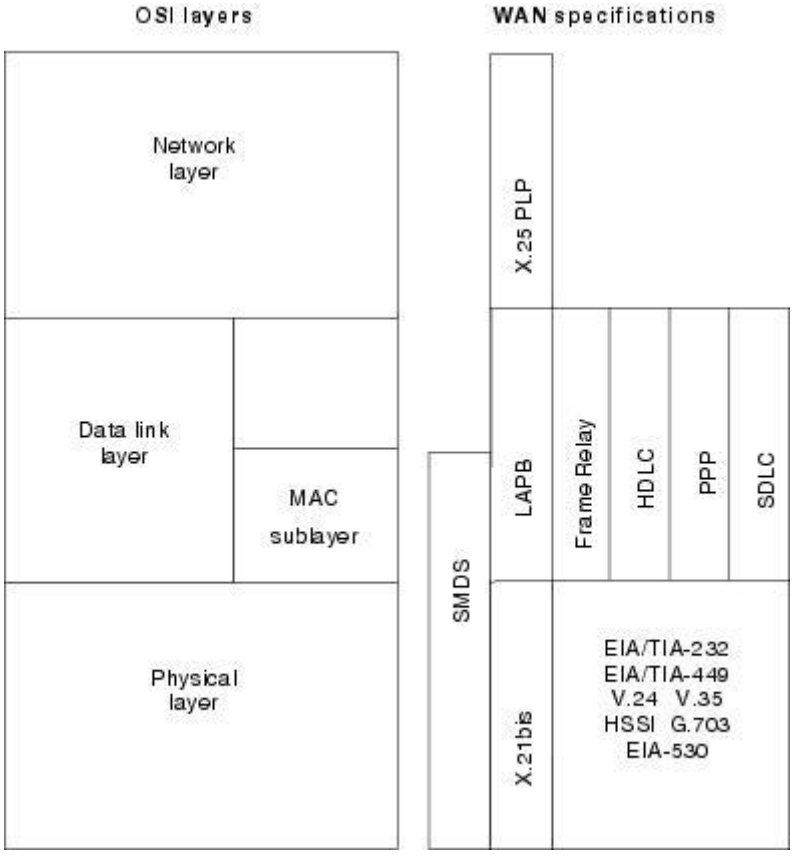
ជំពូកទី៣

WAN (Wide Area Network)

៣-១-សេចក្តីផ្តើមចំពោះ WAN Technologies

តើ WAN ជាអ្វី ?

WAN គឺជាប្រព័ន្ធបញ្ជូន Data តាមប្រព័ន្ធណេតវើកដែលគ្រប់ដណ្តប់លើតំបន់ដែលមានភូមិសាស្ត្រធំ ៗសពេញពិភពលោកតាមរយៈបណ្តាញរបស់ក្រុមហ៊ុនទូរស័ព្ទ ។ WAN technologies ជាទូទៅមានតួនាទីប្រព្រឹត្តិទៅនៅស្រទាប់ទី៣របស់ OSI reference model គឺ physical layer, data link layer, និង network layer ។ រូបភាពខាងក្រោមបង្ហាញពីទំនាក់ទំនងរវាង WAN technologies ដែលគេនិយមប្រើនិង OSI model ។



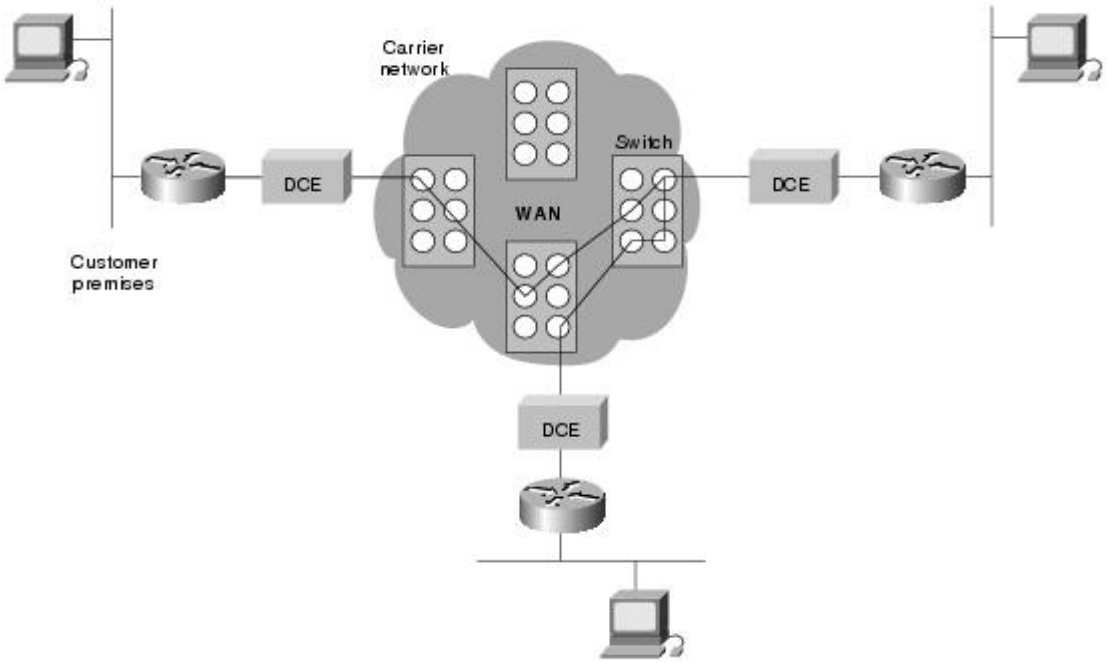
៣-២-ការភ្ជាប់ពីចំណុចមួយទៅចំណុចមួយទៀត

point-to-point link ផ្តល់ឲ្យផ្លូវបញ្ជូនដែលបានបង្កើតឡើងជាមុនតែមួយគត់ពី customer premises តាមរយៈ carrier ណេតវើកតែមួយគត់ដូចជាក្រុមហ៊ុនទូរស័ព្ទដែលភ្ជាប់ចំពោះ remote ណេតវើកមួយ។ បណ្តាញរបស់ Point-to-point ជាទូទៅត្រូវបានជួលពី carrier ហើយគេហៅថា leased lines ។ ចំពោះបណ្តាញ point-to-point carrier បានជ្រើសរើសយកគូរនៃខ្សែនិង hardware ភ្ជាប់មកខ្សែទូរស័ព្ទរបស់អ្នកតែប៉ុណ្ណោះ ។ សៀគ្វីទាំងនេះមានតម្លៃពឹងផ្អែកលើទំហំ bandwidth ដែលត្រូវការនិងចម្ងាយរវាងចំណុចពីរដែលបានភ្ជាប់គ្នា ។ ការភ្ជាប់ពីចំណុច

មួយមកចំណុចមួយទៀតមានតម្លៃថ្លៃជាង shared services ដូចជា Frame Relay។ រូបភាពខាងក្រោមបង្ហាញពីការភ្ជាប់រវាងពីចំណុចមួយមកចំណុចមួយទៀតតាមរយៈWANមួយ ។

៣-២-១-Circuit Switching

Switched circuits អនុញ្ញាតឲ្យការភ្ជាប់ data ដែលអាចចាប់ផ្តើមនៅពេលដែលត្រូវការនិងផ្តាច់ការភ្ជាប់នៅពេលដែលការភ្ជាប់ត្រូវបានបញ្ចប់។ ការងារនេះវាមានលក្ខណៈដូចទៅនិងខ្សែទូរស័ព្ទដែលប្រើសម្រាប់ការប្រាស្រ័យទាក់ទងជាសំឡេង។ Integrated Services Digital Network(ISDN)គឺជាឧទាហរណ៍មួយដ៏ល្អប្រសើរចំពោះ circuit switching។ នៅពេល router មួយមាន data សម្រាប់តំបន់ឆ្ងាយ switched circuit ក៏ត្រូវបានចាប់ផ្តើមជាមួយចំនួននៃ circuit របស់ remote network។ ក្នុងករណីរបស់ ISDN circuits device ជាទូទៅបានហៅមកកាន់របស់លេខទូរស័ព្ទរបស់ remote ISDN circuit ។ នៅពេល Networks ពីរត្រូវបានភ្ជាប់ហើយត្រូវបាន authen ticated វាអាចធ្វើការបញ្ជូន data បាន។ នៅពេលការបញ្ជូន data ត្រូវបានបញ្ចប់ ការហៅទូរស័ព្ទក៏បានផ្តាច់ទៅវិញ។ រូបភាពខាងក្រោមបង្ហាញពីឧទាហរណ៍របស់ប្រភេទនៃសៀគ្វីនេះ។

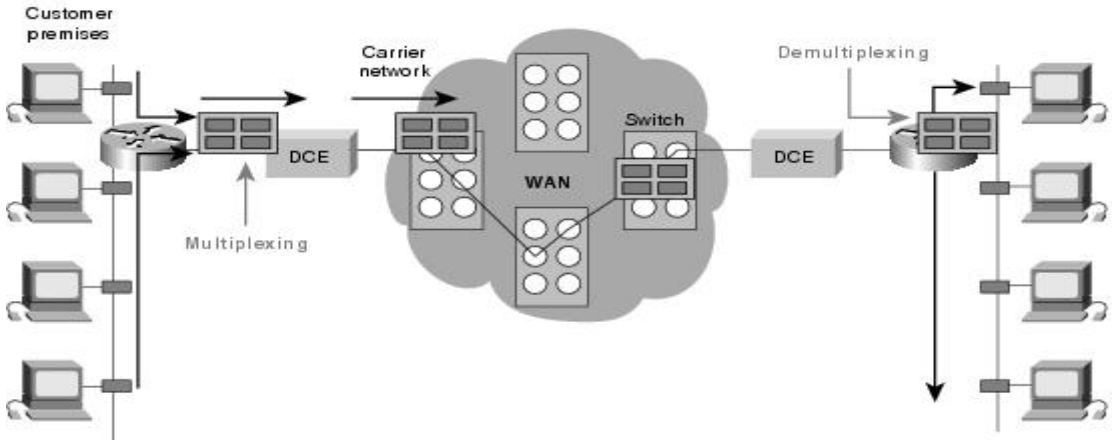


៣-២-២-Packet Switching

Packet switching គឺជា WAN technology មួយដែលអ្នកប្រើប្រាស់បានប្រើ carrier ធនធានរួមគ្នាពីព្រោះថាវាអនុញ្ញាតឲ្យ carrier អាចបង្កើនប្រសិទ្ធភាពនៃការប្រើប្រាស់នៅហេដ្ឋារចនាសម្ព័ន្ធរបស់វា តម្លៃចំពោះ customer ដែលជាទូទៅត្រូវមានលក្ខណៈល្អប្រសើរជាងការភ្ជាប់ពីចំណុចមួយមកចំណុចមួយទៀត។ ក្នុងការបង្កើត packet switching មួយ networks បានភ្ជាប់ជាមួយ carrier របស់ ណោតរើកហើយអតិថិជនភាគច្រើនបានចែលរំលែក

នៅណោតរើករបស់។ បន្ទាប់មក carrier អាចបង្កើត virtual circuits រវាងខាងផ្នែករបស់អតិថិជនដោយ packets របស់ data ត្រូវបានទទួលពី Virtual Circuit មួយតាមរយៈ ណោតរើកមួយផ្សេងទៀត។ ផ្នែកនៃណោតរើករបស់ carrier ដែលត្រូវបានចែករំលែកសំដៅទៅលើសញ្ញាពពក។ ឧទាហរណ៍របស់ packet switching networks រួមមាន Asynchronous Transfer Mode (ATM),Frame Relay, Switched Multimegabit Data Services (SMDS)និង X.25។ រូបភាពខាងក្រោមបង្ហាញពីឧទាហរណ៍របស់ packet-switched circuit។ virtual connections រវាង customer sites គេហៅថាជា virtual circuit ។

រូបភាពនេះបង្ហាញពី Packet Switching ធ្វើការបញ្ជូន Packets ឆ្លងកាត់តាម Carrier ណោតរើក



WAN Virtual Circuits

virtual circuit គឺជា software សៀគ្វីដែលបានបង្កើតក្នុង shared ណោតរើកមួយរវាង ណោតរើកdevices ចំនួនពីរ។ virtual circuits មានដូចជា: switched virtual circuits (SVCs) និង permanent virtual circuits (PVCs)។

SVCs គឺជា virtual circuits ដែលត្រូវបានបង្កើតឡើងជាស្វ័យប្រវត្តតាមតំរូវការហើយបានផ្តាច់ទៅវិញនៅពេលការភ្ជាប់ត្រូវបានបញ្ចប់។ ការប្រាស្រ័យទាក់ទងតាម SVC មួយរួមមានបីដំណាក់កាលគឺការបង្កើតសៀគ្វី ការបញ្ជូន data និងការផ្តាច់សៀគ្វី។ ដំហាននៃការបង្កើតការភ្ជាប់រួមមានបង្កើត virtual circuit រវាងប្រភពដើមនិង devices ដែលជាគោលដៅ។ ការផ្ទេ Data ទាក់ ទងជាមួយការបញ្ជូន data រវាង devices តាម virtual circuit ហើយការផ្តាច់សៀគ្វីទាក់ទងជា មួយការហែក virtual circuit ផ្តាច់រវាងប្រភព devices ដែលជាគោលដៅ។ SVCs ត្រូវបានប្រើក្នុងស្ថានភាពដែលការបញ្ជូន Data រវាង devices ដែលមាន data ប្រែប្រួលធំពីព្រោះថា SVCs បានបង្កើនទំហំbandwidth ដែលបានប្រើដោយសារតែការបង្កើតសៀគ្វីនិងដំណាក់កាលនៃការផ្តាច់ ប៉ុន្តែវាបានកាត់បន្ថយតម្លៃដែលទាក់ទងជា មួយ virtual circuit ដែលមានលក្ខណៈថេរ ។

PVC គឺជា virtual circuit ដែលត្រូវបានបង្កើតឡើងជាអចិន្ត្រៃយ៍ដែលរួមមានការផ្ទេរ data។ PVCs ត្រូវបានប្រើក្នុងស្ថានភាពដែលការផ្ទេរ data រវាង devices មានការថេរ។ PVCs បានកាត់បន្ថយទំហំ bandwidth ដែលប្រើជាមួយការបង្កើតនិងការផ្តាច់នៃ virtual circuits ប៉ុន្តែវាបានកាត់បន្ថយតម្លៃដោយសារតែ virtual circuit មិនប្រែប្រួល។ PVCs ត្រូវបាន configure ជាទូទៅដោយ service provider នៅពេលគេបានកុម្ម៉ង់ service ។

៣-២-WAN Dialup Services

Dialup services គឺជាវិធីសាស្ត្រដែលមានប្រសិទ្ធភាពសម្រាប់ការភ្ជាប់តាម WANs។ ការអនុវត្តនៃ dialup ដែលពេញនិយមមានពីរគឺ dial-on-demand routing (DDR) និង dial backup ។

DDR គឺជាបច្ចេកទេសមួយដែល router មួយអាចចាប់ផ្តើមការហៅនៅលើ switched circuit នៅពេលវាត្រូវការបញ្ជូន data ក្នុងការបង្កើត DDR setup មួយដែល router ត្រូវបាន configureដើម្បីចាប់ផ្តើមដំណើរការហៅនៅពេលដែលជួបជាមួយលក្ខ័ខ័ណ្ឌវិនិច្ឆ័យដូចជាប្រភេទនៃ ណេតវើកចារចរណ៍ ពិសេសដែលត្រូវការសម្រាប់បញ្ជូន។ នៅពេលបានភ្ជាប់ ចារចរណ៍ ត្រូវបានឆ្លងកាត់តាមបណ្តាញ។ ការ configure ចំពោះ router បានកំណត់រយៈពេល សម្រាកដែលបានប្រាប់ Router ឲ្យបោះចោលការភ្ជាប់នៅពេលសៀគ្វីត្រូវបានផ្អាកក្នុងរយៈពេលមួយ ។

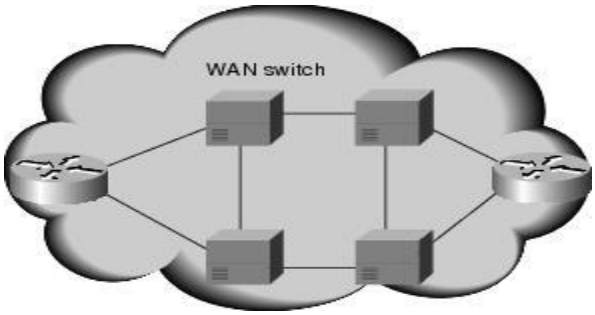
Dial backup គឺជាវិធីសាស្ត្រមួយទៀតនៃការ Configure DDR។ ទោះបីជាយ៉ាងណាក៏ដោយក្នុង dial backup switched circuit ត្រូវបានប្រើប្រាស់ដើម្បីផ្តល់នៅ backup service សម្រាប់ប្រភេទនៃសៀគ្វីមួយទៀតដូចជា Point-to-point ឬ Packet Switching។ Routerត្រូវបាន Configure ដូច្នេះនៅពេលវាដឹងថាវាខូចនៅលើសៀគ្វីទី១បណ្តាញ Dial backupក៏បានចាប់ផ្តើម។ បន្ទាប់មកបណ្តាញ Dial backup ក៏បានភ្ជាប់សម្រាប់ WAN រហូតទាល់តែសៀគ្វីទី១ត្រូវបានជួសជុលឡើងវិញ។ នៅពេលមានបញ្ហានេះកើតឡើងការភ្ជាប់សម្រាប់ Dial backup ក៏ត្រូវបានផ្តាច់ទៅវិញ ។

WAN Devices

WANs ប្រើប្រភេទនៃDevicesផ្សេងៗគ្នាដែលបានកំណត់សម្រាប់ WAN។ WAN switches, Access Servers, Modems, CSU/DSUs និង ISDN terminal Adapters ត្រូវបានលើកយកមកសិក្សាក្នុងផ្នែកខាងក្រោម។ Devices ប្រភេទផ្សេងៗទៀតដែលយកមកប្រើមានដូចជា Routers ATM Switches និង Multiplexers ។

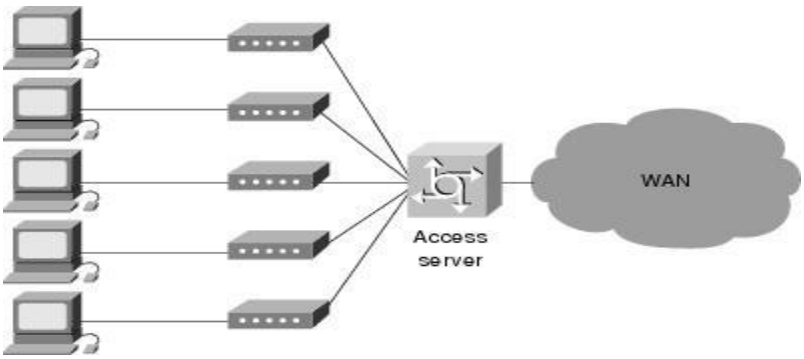
៣-២-១-WAN Switch

WAN switch គឺជា internetworking devices ដែលមានច្រើន ports។ Devices ទាំងនេះមានដូចជា Frame Relay, X.25 និង SMD ហើយដំណើរការនៅស្រទាប់ទី២នៃ OSI reference Model។ រូបភាពខាងក្រោមបង្ហាញពី Routersពីរដែលស្ថិតនៅឆ្ងាយនៃ WAN ដែលត្រូវបានភ្ជាប់ជាមួយ Switches។ Routers ពីរស្ថិតនៅឆ្ងាយផ្នែកទាំងសងខាងរបស់ WANដែលភ្ជាប់គ្នាដោយ WAN Switches



៣-២-២-Access Server

access server ដើរតួនាទីជាចំណុចកណ្តាលសម្រាប់ Dial-in និង dial-out។ រូបភាពខាងក្រោមបង្ហាញពី Access Server មួយកំពុងប្រើការភ្ជាប់តាម Dial-out ជាមួយ WAN មួយ ។

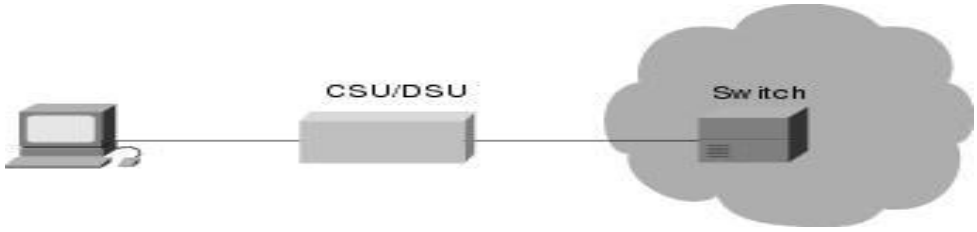


៣-២-៣-Modem

modem គឺជាឧបករណ៍សម្រាប់បកប្រែ digital និង Analog Signal ដែលធ្វើឲ្យ Data អាចបញ្ជូនតាមបណ្តាញទូរស័ព្ទ។ នៅខាងប្រភពដើម Digital Signal ត្រូវបានបំប្លែងជាទម្រង់មួយសម្រាប់បញ្ជូនតាមប្រព័ន្ធ Analog។ នៅឯឧបករណ៍ទទួលវិញanalog signal ត្រូវបានបំប្លែងជា Digital Signal វិញ។ នៅឯគោលដៅ analog Signal ត្រូវបានត្រឡប់មកជាDigitalវិញ ។ រូបភាពខាងក្រោមបង្ហាញពីការភ្ជាប់ពី Modem ជាមួយ Modem តាមរយៈ WAN ។

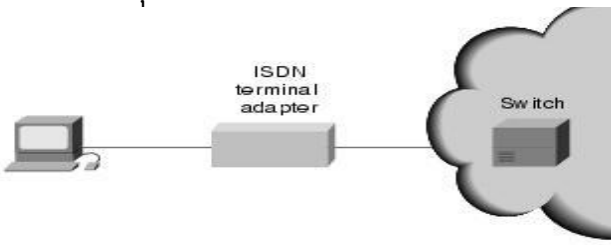
៣-២-៤-CSU/DSU

channel service unit/digital service unit (CSU/DSU) គឺជា digital interface device ដែលបានប្រើសម្រាប់ភ្ជាប់ Router ជាមួយ Digital Circuit ដូចជាT1 ។ CSD/DSU ក៏ផ្តល់ឲ្យនៅ Signal សម្រាប់ការប្រាស្រ័យទាក់ទងជាមួយឧបករណ៍ទាំងនេះ។ រូបភាពខាងក្រោមបង្ហាញពីការកំណត់ទីតាំងរបស់ CSU/DSU ក្នុងការប្រើ WAN ។



៣-២-៥-ISDN Terminal Adapter

ISDN terminal adapterគឺជាឧបករណ៍ដែលប្រើសម្រាប់ភ្ជាប់ជាមួយ ISDN Basic Rate Interface (BRI) ជាមួយ interface ផ្សេងទៀតដូចជា EIA/TIA-232 លើ Router មួយ។ Terminal Adapter គឺជា ISDN modem ពីព្រោះថាវាមិនបំប្លែង Analog មកជា Digital Signal នោះទេ។ រូបភាពខាងក្រោមបង្ហាញពីការកំណត់ទីតាំងរបស់ Terminal Adapter ក្នុង ISDN ។



សំណួរម្តឹកឡើងវិញ

Q—តើប្រភេទនៃ WAN circuits មានអ្វីខ្លះ ?

A—Point-to-point, packet-switched, and circuit-switched.

Q—តើ DDR គឺជាអ្វីហើយវាខុសគ្នាពី dial backupត្រង់ណាខ្លះ ?

A—DDR គឺជា dial-on-demand routing ។ DDR សម្រាប់ភ្ជាប់ជាមួយ remote site នៅពេលដែលចរាចរណ៍ណោតរើកត្រូវការបញ្ជូន ។ Dial backup ប្រើប្រភេទនៃ Service ដូចគ្នា ប៉ុន្តែចំពោះ backup វិញត្រូវបានប្រើជំនួស primary circuit ។ នៅពេលដែល primary circuit ខូច dial backup line ត្រូវបានចាប់ផ្តើមរហូតទាល់តែ primary circuit ត្រូវបានជួសជុលឡើងវិញ ។

Q—តើ CSU/DSU ត្រូវបានប្រើប្រាស់សម្រាប់ធ្វើអ្វី ?

A—A CSU/DSU សម្រាប់ផ្តល់ interfaces ឲ្យ router ដែលមាន digital line ដូចជា T1 ។

Q—តើ modem និង ISDN terminal adapter ខុសគ្នាត្រង់ចំណុចណាខ្លះ ?

A—A modem បំប្លែង digital signals ឲ្យមកជា analog ដើម្បីធ្វើការបញ្ជូនតាមប្រព័ន្ធទូរស័ព្ទ ។ ចំណែកឯ ISDN circuits វិញគឺជា digital ។ ដូច្នេះមិនត្រូវការបំប្លែងនោះទេ ។

ជំពូកទី៤

IPv40 គឺជា Protocol មួយដែលគេពេញនិយមប្រើប្រាស់សព្វថ្ងៃនេះ។ IPv4.0 ត្រូវបានបង្កើតឡើងនៅក្នុងទសវត្សរ៍១៩៧0s។ ចំពោះ IPv6.0 វិញត្រូវបានបង្កើតឡើងប្រើប្រាស់ដោយសារតែមានការខ្វះខាតនៃ IP address ពីព្រោះមានឧបករណ៍ជាច្រើនត្រូវបានប្រើប្រាស់ IPv4.0អស់។

IPv6.0 ត្រូវបានវិស្វករនឹងអ្នកបង្កើត Software ប្រើប្រាស់ដើមទសវត្ស១៩៩0s ។ IP address ត្រូវបានពង្រីកពី 32 bits មកជា 128 bits address ។

- IPV6 addresses មានប្រវែងស្មើនិង 128 bits ដែលមានទម្រង់ជា: x:x:x:x:x:x:x:x ដែលលេខប្រព័ន្ធគោល ១៦មានតម្លៃលេខពី(0000-FFFF) ។
- នៅក្នុងករណីខ្លះ IPV6 addresses មានសុទ្ធតែលេខសូន្យទាំងអស់ដូចជា 0000:0000:0000: 0000:0000:0000: 0000:1 ត្រូវបានគេសរសេរជា 0:0:0: 0:0:0: 0:1 ។ វាបានបង្ហាញពីក្រុមនៃ bit លេខសូន្យដោយប្រើនិមិត្តសញ្ញា“::” ។ ដូច្នេះគេអាចសរសេរដោយអក្សរកាត់ជា“::1” ។ ដោយប្រើសញ្ញា “::” IPV6 អាចកំណត់ពីចំនួនតម្លៃនៃលេខដែលបានបាត់គឺជាលេខសូន្យទាំងអស់។ ទោះបីជាយ៉ាងណាក៏ដោយបើសញ្ញា “::” ត្រូវបានប្រើច្រើនជាងមួយដង វាមិនអាចប្រាប់ឲ្យដឹងពីចំនួន bit នៃលេខសូន្យដែលបានបាត់នោះទេ ។

៤-១-ការពិពណ៌នាអំពី IPv 6.0 Packet Header

Header មានប្រវែង 40 bits ហើយការរៀបចំមាន Version, Class, Flow Label, Payload, Length, Next Header, Hop Limit, Source address, Destination Address, Data និង Payload fields ។

Hexadecimal “Hex”

0 1 2 3 4 5 6 7 8 9 A B C D E F

A = 10

B = 11

C = 12

D = 13

E = 14

F = 15

លេខ0ដល់៩ក្នុងប្រព័ន្ធគោល១0ស្មើនឹង0ដល់៩ក្នុងប្រព័ន្ធគោល១៦

៤-២-ការពិពណ៌នាអំពី Address

ចូរពិនិត្យទៅលើឧទាហរណ៍របស់ IPv6 address ។ Address ត្រូវបានរៀបចំតាម៨ផ្នែកដែលចែកដាច់ពីគ្នាដោយសញ្ញា “:” ។ ផ្នែកនីមួយៗមានប្រវែង 16 bits ។ ដូច្នេះ IPv6.0 មានប្រវែងស្មើនិង 128 bits (16 bits x 8) ។

Address មានទម្រង់ជា:

n:n:n:n:n:n:n:n ដែល n ស្មើ៤ខ្ទង់ជាគោល១៦

ឧ 1080:0:0:0:8:800:200C:417A ជា Unicast

FF01:0:0:0:0:0:0:101 ជា multicast address

៤-២-១-Broadcasting Methods

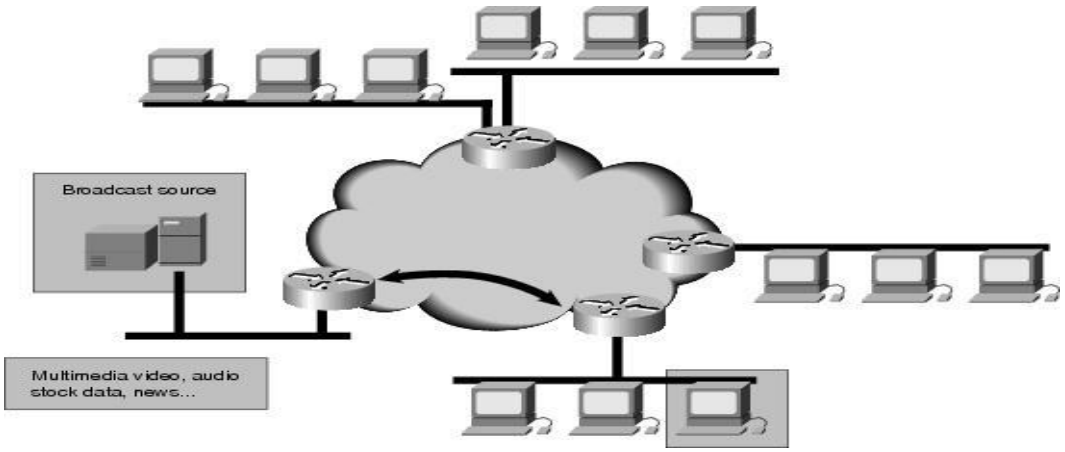
នៅក្នុង IPv6.0 មានវិធីនៃការ Broadcast ថ្មីដូចជា:

- Unicast
- Multicast
- Anycast

៦-២-១-១-Unicast

Unicast គឺជាការប្រាស្រ័យទាក់ទងរវាង Host តែមួយគត់ជាមួយឧបករណ៍ទទួលតែមួយគត់។ Packet ត្រូវបានបញ្ជូនទៅ Unicast address ហើយត្រូវបានទទួលតែចំពោះ Interface ដែលមាន Address កំនត់ក្នុងនោះប៉ុណ្ណោះ។

	Long format	Shortcut format
Unicast	1070:200:0:0:0:900:300C:618A	1070:200::900:300C:618A



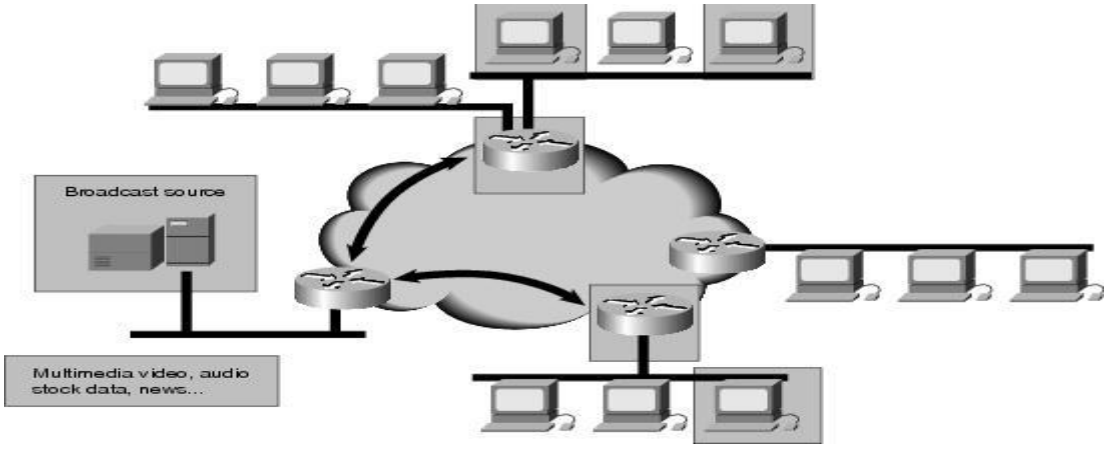
រូបភាព-១-Unicast បញ្ជូន Packets ទៅឲ្យ Interface ច្បាស់លាស់

៤-២-១-២-Multicast

Multicast គឺជាការប្រាស្រ័យទាក់ទងរវាង Host តែមួយគត់ជាមួយឧបករណ៍ទទួលជាច្រើន។ Packets ត្រូវបានបញ្ជូនទៅឲ្យគ្រប់ Interface ទាំងអស់ដែលបានកំនត់ដោយ Address មាននៅក្នុងនោះ។

	Long format	shortcut
Multicast	FF01:0:0:0:0:0:0:100	FF01::100

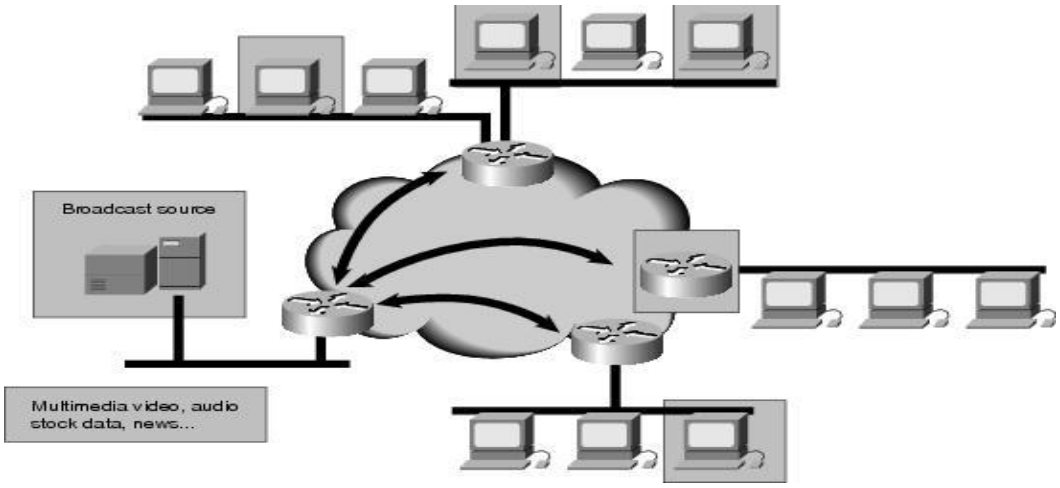
- FF01::1 សម្រាប់ nodes ទាំងអស់ដែលស្ថិតនៅក្នុង node-local scope
- FF02::1 សម្រាប់ nodes នៅលើ local link
- FF01::2 សម្រាប់ routers នៅក្នុង node-local scope.
- FF02:: 2 សម្រាប់ routers នៅលើ link-local scope.
- FF05::2 សម្រាប់ routers នៅក្នុង site



រូបភាព-២ Multicast បញ្ជូន Packets ទៅឲ្យ Subnet និងឧបករណ៍ដែលបានកំនត់ទុកដើម្បីសម្រាប់ Multicast Packets

៤-២-១-៣-Any cast

Packets ដែលបានបញ្ជូនចំពោះ anycast address ឬបណ្តុំនៃ Addresses ត្រូវបានទទួលដោយ Interface ដែលស្ថិតនៅជិតបំផុតដែលបានកំនត់ដោយ Address នោះ។ Anycast គឺជាការប្រាស្រ័យទាក់ទងរវាងឧបករណ៍បញ្ជូនតែមួយជាមួយបណ្តុំនៃ Address ។



រូបភាព-៣- Anycast បញ្ជូន Packets ទៅឲ្យបណ្តុំនៃ Interface ហើយអាចមានឧបករណ៍ចុងបញ្ចប់និង Routers

៤-៣-IP Multicast Addresses

Multicast addresses កំនត់ច្បាស់លាស់ពីក្រុមនៃ IP host ដែលបានចូលរួមក្នុងក្រុមហើយចង់ទទួលចរាចរណ៍ដែលបានបញ្ជូនឲ្យចំពោះក្រុមនេះ។

IP class D address

IANA (Internet Assigned Number Authority) គ្រប់គ្រងក្នុងការកំនត់ IP multicast address។ គេបានកំនត់ Class D address ប្រើជា IP multicast។ ដូច្នេះ IP multicast group addresses ចាប់ផ្តើមពី 224.0.0.0 → 239.255.255.255 ។

Reserved Link Local Addresses

IANA បានបម្រុង Address ចាប់ពី 224.0.0.0 ដល់ 224.0.0.255 សម្រាប់ ណេតវើកprotocol នៅលើ LAN segement ណាមួយ។ Packets ប្រើជាមួយ Addresse ទាំងនេគឺ Router មិនអាចបញ្ជូនបន្តនោះបានទេវាស្ថិតនៅក្នុងផ្នែក LAN មួយ។ វាត្រូវបញ្ជូនជាមួយTTL(Time-to-Live)នៃ1s។ ណេតវើកprotocolsប្រើ addresses ទាំងនេះសម្រាប់រក Router ជាស្វ័យប្រវត្តិហើយសម្រាប់ទំនាក់ទំនងជាមួយ Routing ព័ត៌មានដែលបានកំនត់។ ឧទាហរណ៍ OSPF ប្រើ 224.0.0.5 និង224.0.0.6 ដើម្បីផ្លាស់ប្តូរព័ត៌មានអំពីលក្ខណៈរបស់ Link។ តាមតារាងខាងក្រោមបង្ហាញពី Address ដែលគេស្គាល់:

Table Link Local Addresses	
Address	ការប្រើប្រាស់
224.0.0.1	គ្រប់ប្រព័ន្ធទាំងអស់នៅលើ subnet នេះ
224.0.0.2	គ្រប់ routers ទាំងអស់នៅលើ subnet នេះ
224.0.0.5	OSPF routers
224.0.0.6	OSPF designated routers
224.0.0.12	DHCP server/relay agent

Global Scoped Address

Address ចាប់ផ្តើមពី 224.0.1.0ដល់ 238.255.255.255 ដែលគេហៅថា globally scoped address។ គេប្រើប្រាស់វាសម្រាប់បញ្ជូនទិន្នន័យតាមអង្គការនិងឆ្លងកាត់តាមប្រព័ន្ធអិនធឺណេត។ Address ត្រូវបានបម្រុងទុកសម្រាប់ប្រើប្រាស់ដោយ multicast application តាមរយៈ IANA។ ឧទាហរណ៍ 224.0.1.1 ត្រូវបានប្រើប្រាស់សម្រាប់ ណេតវើកTime Protocol (NTP)។

Limited Scoped Address

Addresses ចាប់ផ្តើមពី 239.0.0.0 រហូតដល់ 239.255.255.255 បានកំនត់ address សម្រាប់ scope address Address ទាំងនេះត្រូវបានកំនត់ដោយ RFC 2365 ដោយប្រើសម្រាប់ local group ឬអង្គការ។

Global Addressing

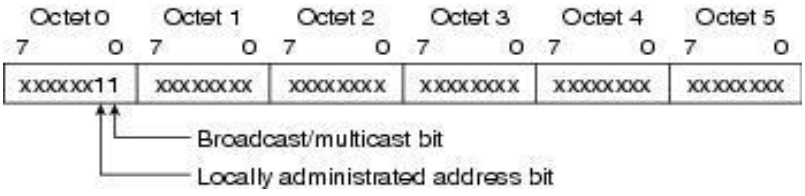
RFC 2770 បានស្នើ address 233.0.0.0/8 សម្រាប់បម្រុងទុកឲ្យអង្គការដែលបានបម្រុង AS number ។ AS number របស់ domain ត្រូវបានភ្ជាប់ជាមួយ Octet ទី២និងទី៣នៃ233.0.0.0/8 ។ ឧទាហរណ៍ As 62010 ត្រូវបានសរសេរជាគោល១៦គឺF23A ។ យើងញែកវាជាពីរគឺ F2 និង 3A នោះយើងទទួលបាន 242 និង58 ជាគោល១០ ។ វាបានផ្តល់ឲ្យយើងនូវ Subnet មួយនៃ 233.242.58.0 ដែលត្រូវបានបម្រុងទុកសម្រាប់ As 62010 ។

Layer 2 Multicast Addresses

តាមធម្មតា NICs ដែលស្ថិតនៅលើផ្នែកមួយនៃ LAN មួយនិងទទួលតែ Packets ដែលមានកំនត់គោលដៅសម្រាប់ MAC ឬ Broadcast MAC address ។ Address ខ្លះត្រូវបានប្រើប្រាស់ជា Multicast ដែលធ្វើឲ្យ Hosts អាចទទួលបាននូវ Packets ហើយគេអាចបែងចែកនូវភាពខុសគ្នានៃ Multicast group ។

តែសំណាងល្អដែលការកំនត់ IEEE LAN បានបង្កើតឡើងជាបណ្តោះអាសន្នសម្រាប់ធ្វើការបញ្ជូននៃ Broadcast ឬ Multicast packets ។

នៅក្នុងស្តង់ដា 802.3 standard bit 0 នៃ octet ទី១ ត្រូវបានប្រើដើម្បីបង្ហាញពី broadcast/Multicast bit ក្នុង Ethernet frame ។



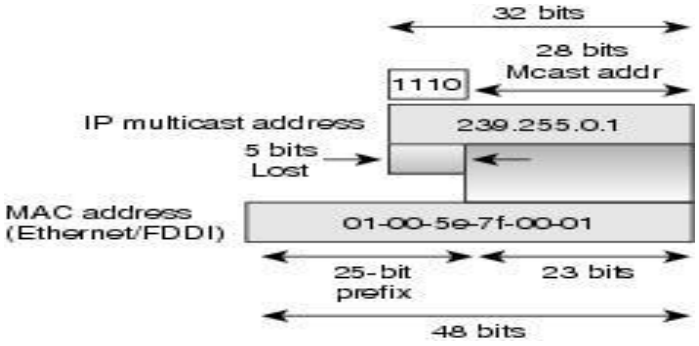
រូបភាព-៣-IEEE 802.3 MAC Address Format

bit នេះបង្ហាញជា Frame ដែលត្រូវបានកំនត់ជាគោលដៅចំពោះក្រុមណាមួយនៃ Hosts ឬគ្រប់ Hosts ទាំងអស់នៅលើ Network ។ ចំពោះ broadcast address គឺ 0xFFFF.FFFF. FFFF ។

Ethernet MAC Address Mapping

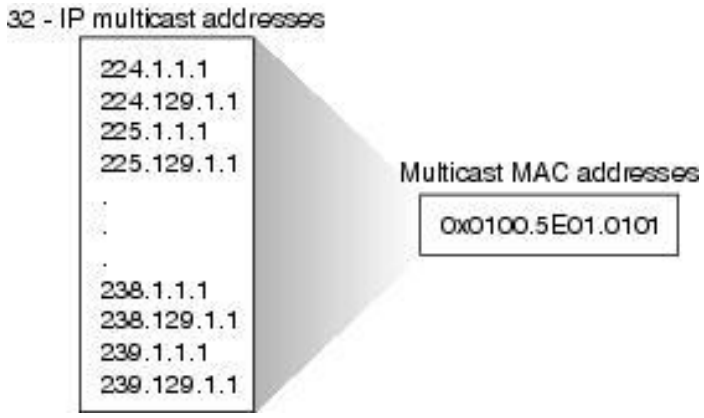
IANA មានបណ្តុំនៃ Ethernet MAC address ផ្ទាល់ដែលចាប់ផ្តើមជាមួយ 01:00:5E (គោល 16) ។ ពាក់កណ្តាលនៃបណ្តុំនេះត្រូវបានកំនត់ប្រើជា Multicast address ។ នេះបង្កើតបាននូវ Ethernet MAC address ចាប់ពី 0100.5e00.0000 រហូតដល់ 0100.5e7f.ffff ។

ការកំនត់បែបនេះអនុញ្ញាតឲ្យ 32 bits ក្នុង Ethernet Address ដើម្បីបំពេញចំពោះ IP multicast group address ។



រូបភាព-៤: ការកំនត់ឲ្យត្រូវនៃ IP Multicast ជាមួយ Ethernet/FDDI MAC Address

ដោយសារតែ 5 bits ខាងលើនៃ IP multicast address ត្រូវបានបោះចោលក្នុងការកំនត់នេះ ដូច្នេះ address មិនមែនតែមួយប៉ុណ្ណោះ។ តាមពិត 32 multicast group IDs ខុសៗគ្នាដែលត្រូវគ្នាជាមួយ Ethernet address។



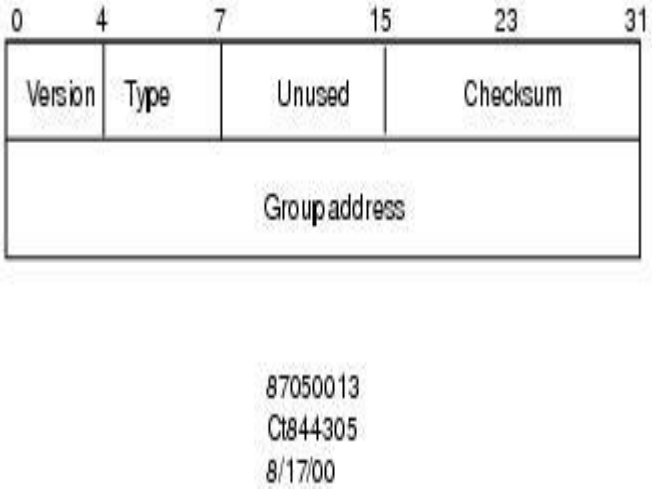
រូបភាព-៥- MAC Address Ambiguities

៤-៤-Internet Group Management Protocol (IGMP)

IGMP ត្រូវបានគេប្រើដើម្បីចុះបញ្ជី Host នីមួយៗជាស្វ័យប្រវត្តក្នុង Multicast group នៅលើ LAN ណាមួយ។ Host អាចសម្គាល់ភាពជាសមាជិកបានដោយបញ្ចូល IGMP message ទៅឲ្យ local multicast router។ ក្រោម IGMP routerស្គាល់ចំពោះ IGMP message ហើយក្នុងរយៈពេលច្បាស់ លាស់វាបញ្ជូនចេញនូវការសាកសួរដើម្បីរកឲ្យឃើញក្រុមណាមួយដែលសកម្មឬមិនសកម្មនៅលើ Subnet ។

IGMP Version 1

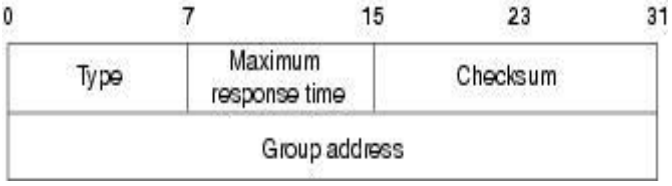
RFC 1112 កំនត់ច្បាស់លាស់សម្រាប់ IGMPV 1។



រូបភាព-៦- IGMP Version 1 Packet Format

IGMP Version 2

RFC 2236 កំនត់ច្បាស់លាស់សម្រាប់ IGMPV2 ។



រូបភាព-៧- IGMPv2 Message Format

IGMP V2 ធ្វើការដូច IGMPv1 ដែរ។ ភាពខុសគ្នាជាចំបងគឺមាន Leave Group Messag មួយ ។

ឥឡូវនេះ Hosts អាចធ្វើការទាក់ទងជាមួយ local multicast router ដែលការយកចិត្តទុកដាក់គឺចាកចេញពីក្រុម។ បន្ទាបមក Router បញ្ជូនចេញនូវការសាកសួរហើយកំនត់ថាតើមាន Host នៅសល់ដែលពេញចិត្តទទួលចរាចរណ៍។ បើគ្មានការឆ្លើយតបទេ Router កំនត់រយៈពេលហួសកំនត់ឈប់បញ្ជូនបន្ត។ ការធ្វើបែបនេះកាត់បន្ថយយ៉ាងច្រើននូវ Leave latency បើប្រៀបធៀបជាមួយ IGMPv1 ។ ចរាចរណ៍ដែលមិនត្រូវការនឹងមិនចាំបាច់ត្រូវបានបញ្ឈប់យ៉ាងឆាប់រហ័ស។

៤-៥-Multicast នៅក្នុងមជ្ឈដ្ឋាន layer 2 Switching

លក្ខណៈតាមលំនាំដើមសម្រាប់ Layer 2 Switch គឺត្រូវបញ្ជូនបន្តនូវចរាចរណ៍ Multicast ទាំងអស់ទៅគ្រប់ Ports ដែលជារបស់ LAN គោលដៅនៅលើ Switch ។ នេះជាលក្ខណៈរបស់ Switch ដែលបានកំនត់ចរាចរណ៍ចំពោះ Port ដែលត្រូវការទទួលទិន្នន័យ។

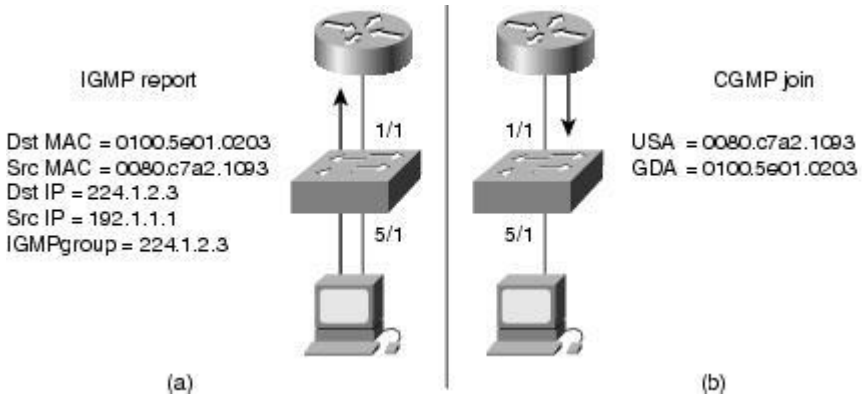
វិធីសាស្ត្រពីរដែលប្រើជាមួយ Multicast ក្នុង Layer 2 Switching គឺ Cisco Group Management Protocol(CGMP) និង IGMP ។

៤-៥-9-Cisco Group Management Protocol (CGMP)

CGMP គឺជា Protocol របស់ Cisco ដែលអនុញ្ញាតឲ្យ Catalyst Switch កំនត់ព័ត៌មាន IGMP នៅលើ Cisoc Router ដើម្បីធ្វើឲ្យ Layer 2 សម្រេចបញ្ជូនបន្ត ។ CGMP ត្រូវ Configure នៅលើ multicast routers និងលើ Layer 2 swicthes ។ ជាលទ្ធផលជាមួយចរាចរណ៍ CGMP IP multicast ត្រូវទទួលតែចំពោះ Catalyst Switch Port ដែលប្រើសម្រាប់ចរាចរណ៍ ។ Ports ផ្សេងៗទៀតគ្មានចរាចរណ៍និងមិនទទួលវា ។ មូលដ្ឋានគ្រឹៈនៃ CGM បង្ហាញក្នុងរូបភាពខាងក្រោម ។ នៅពេលដែល Host មួយចូលរួមក្នុង multicast group(Port A) វា multicast នូវ IGMP membership message ឲ្យគ្រប់គោលដៅក្រុម(224.1.2.3)ក្នុងឧទាហរណ៍ ។ របាយការណ៍ IGMP ត្រូវបានបញ្ជូនឆ្លងកាត់ Switch ទៅកាន់ Router ដើម្បីឲ្យ IGMP ដំណើរការ ។ Router (ត្រូវតែមាន CGMP បើកនៅលើ interface នេះ)ទទួលរបាយការណ៍ IGMP ហើយដំណើរការវាតាមធម្មតា ប៉ុន្តែវាបង្កើត CGMP មួយដើម្បីចូលរួមសារនិងបញ្ជូនវាទៅឲ្យ switch ។

Switch ទទួល CGMP នេះចូលក្នុងសារហើយបន្ទាប់មកបន្ថែម Port ឲ្យ CAM(content addressable memory) table សម្រាប់ multicast group ។ ជាលទ្ធផលចរាចរណ៍ត្រូវបានដឹកនាំដោយផ្ទាល់ទៅឲ្យ multicast

group នេះនឹងត្រូវបញ្ជូនបន្តចេញតាម Port ចំពោះ Host នោះ។ Port របស់ Router ក៏ត្រូវបានបន្ថែមផងដែរ ចំពោះ multicast group។ Multicast routers ត្រូវបានបន្ថែមផងដែរចំពោះ multicast group។ Multicast routers ត្រូវតែស្តាប់ចំពោះចរាចរណ៍ Multicast ទាំងអស់ចំពោះ Group នីមួយៗពីព្រោះ IGMP គ្រប់គ្រងសារ ដែលត្រូវបានបញ្ចូលជាលក្ខណៈចរាចរណ៍ multicast ផងដែរ។ ជាមួយ CGMP Switch ត្រូវតែស្តាប់ចំពោះ CGMP ដែលចូលរួមនិង CGMP ដែលចាកចេញសារពី Router ។ ផ្នែកនៅសល់របស់ចរាចរណ៍ multicast ត្រូវបាន បញ្ជូនបន្តដោយប្រើ CAM table របស់វាដូចគ្នានឹងវិធីសាស្ត្រដែល Switch ត្រូវបានបង្កើតឡើង។



រូបភាព-៨- Basic CGMP Operation

IGMP Snooping

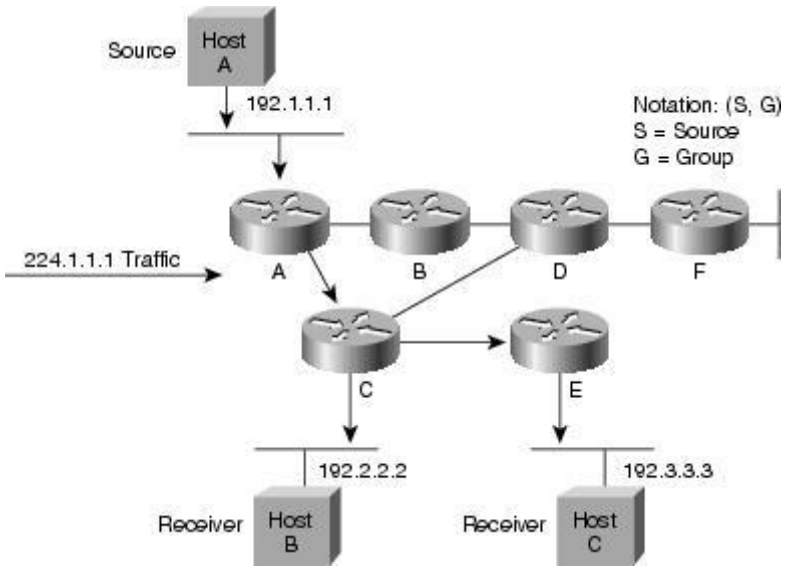
IGMP Snooping ទាមទារឲ្យ LAN Switch ត្រួតពិនិត្យនូវព័ត៌មានរបស់ Layer 2 នៅក្នុង IGMP packets ដែលត្រូវបានបញ្ជូនរវាង Host និង Router។ នៅពេលដែល Switch ដឹងពីរបាយការនៃ IGMP host ពី Host មួយ សម្រាប់ multicast group, switch បន្ថែម Port របស់ Host ទៅនិង multicast table។ នៅពេលដែល Switch បានដឹងពី IGMP ចាកចេញពី Host មួយវាក៏លុប Port របស់ Host ពីតារាង។

៤-៦-Multicast Destination Trees

Router ដែលមានសមត្ថភាពបង្កើត Tree ជាគោលដៅដែលគ្រប់គ្រងផ្លូវបញ្ជូនដែលចរាចរណ៍ IP multicast ឆ្លងកាត់តាម ណេតវើកដើម្បីទទួលយកចរាចរណ៍ឲ្យគ្រប់ឧបករណ៍ទទួល។ មាន Multicast destination trees ប្រភេទគឺ Source trees និង Share trees។

Source Trees

ជាទម្រង់ងាយរបស់ multicast destination tree គឺ Source tree ដែល Root របស់វាគឺជាប្រភពនៃ multicast tree ហើយសាខារបស់វាបង្កើតបានជា spanning tree តាមរយៈ ណេតវើក មកកាន់ឧបករណ៍ទទួល។ ដោយសារតែ Tree នេះប្រើ Shortest Path តាមរយៈ ណេតវើកដែលគេហៅវាថា Shortest Path Tree(SPT) ។



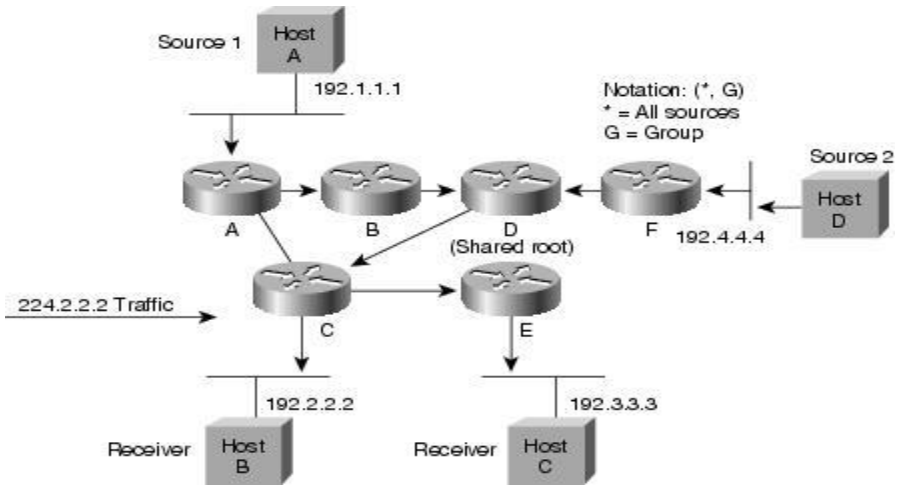
រូបភាព-៩- Host A Shortest Path Tree

(S,G) គេអាចថា " S comma G" ដែល S ជា IP address របស់ប្រភពដើមហើយ G ជា multicast group address។ ការប្រើសញ្ញាសម្គាល់នេះ SPT ក្នុងឧទាហរណ៍គឺ(192.1.1.1, 224.1.1.1)។

សញ្ញាសម្គាល់(S,G) មានន័យថា SPT មានស្រាប់សម្រាប់ប្រភពដើមនីមួយៗដែលកំពុងបញ្ជូនឲ្យក្រុមនីមួយៗដែលត្រឹមត្រូវ។ ឧទាហរណ៍បើ Host B កំពុងបញ្ជូនចរាចរណ៍ឲ្យ Group 224.1.1.1 ហើយ Host A និង C គឺជាឧបករណ៍ទទួលនោះ (S,G) SPT មានសញ្ញាសម្គាល់(192.2.2.2, 224.1.1.1)។

Shared Trees

មិនដូចជា Source Tree ទេដែលមាន Root នៅឯប្រភពដើម Share tree ប្រើ root តែមួយគត់ដែលបានដាក់ត្រង់ចំណុចណាមួយក្នុង ណេតវើក។ Shared root នេះគេហៅថា rendezrous point (RP)។ រូបខាងក្រោមបង្ហាញពី Shared Tree សម្រាប់ Group 224.2.2.2 ជាមួយ Root ដែលស្ថិតនៅឯ Router D។ នៅពេលយើងជ្រើសរើសយក Shared tree នោះ Source ត្រូវតែបញ្ជូនចរាចរណ៍របស់ពួកគេឲ្យ root ហើយបន្ទាប់មកចរាចរណ៍ត្រូវបញ្ជូនតាម Shared tree ទៅឲ្យឧបករណ៍ទទួលនីមួយៗ។



រូបភាព-១0-Shared Distribution Tree

នៅក្នុងឧទាហរណ៍នេះចរាចរណ៍ Multicast ពីប្រភពដើម Host A និង D ធ្វើដំណើរមក root (Router D) ហើយបន្ទាប់មកចុះមក Shared tree មកកាន់ឧបករណ៍ទទួលពីរគឺ Host B និង C។ ដោយសារតែប្រភពដើមទាំងអស់ស្ថិតនៅក្នុង multicast group ប្រើ Shared tree រួមគ្នា(*,G) អានថា “Star comma G” តំណាងឲ្យ tree។ ក្នុងករណីនេះ * មានន័យថាគ្រប់ប្រភពដើមទាំងអស់និង G តំណាងឲ្យ multicast group។ ដូច្នោះ Shared tree បង្ហាញក្នុងរូបខាងលើ (4.8) ត្រូវសរសេរជា (*, 224.2.2.2)។ ទាំង SPT និង Shared tree ជា loop-free។ Messages មាន replicate តែចំពោះ tree ដែលមានសាខា។

សំណួរម្លឹកឡើងវិញ

Q—តើ Standard នៃ IP address ដែលកំពុងប្រើបច្ចុប្បន្នជា version ប៉ុន្មាន *?*

A—IPv4.

Q—តើមានហេតុផលអ្វីខ្លះដែលគេបង្កើត IPv6 Addres ឡើង *?*

A—ហេតុផលជាចំបងនៃការបង្កើត IPv6 ឡើងគឺ IPv4 ដែលកំពុងប្រើនិងត្រូវបានប្រើអស់។ ដូច្នេះ IPv6 គឺជាដំណោះស្រាយ ប៉ុន្តែវានៅមិនទាន់ក្លាយជា Standard នៅឡើយនោះទេ។

Q—តើវាបានពង្រីកប៉ុន្មាន bits *?*

A—វាបានពង្រីកពី 32 bits មក 128 bits address

Q—តើការពង្រីកនោះវាផ្តល់បានផលប្រយោជន៍អ្វីខ្លះ *?*

A—វាបានផ្តល់ឲ្យនូវ unicast និង broadcasting method ថ្មី។ ចំពោះ Address ដែលបានពង្រីកថ្មីនេះបានប្រើសញ្ញា":" ជំនួសឲ្យ “.”។

Q—តើ broadcast method ថ្មីមាននៅក្នុង IPv6 អ្វីខ្លះ *?*

A—Unicast, multicast, and anycast.

Q—តើអ្វីទៅជា *unicast?*

A— Unicast គឺជាការប្រាស្រ័យទាក់ទងរវាងឧបករណ៍បញ្ជូនតែមួយនិងឧបករណ៍ទទួលតែមួយ

Q—តើអ្វីទៅជា *multicast?*

A—Multicast គឺជាការប្រាស្រ័យទាក់ទងរវាង Host តែមួយគត់និងឧបករណ៍ទទួលជាច្រើន

Q—តើ anycast គឺជាអ្វី *?*

A—Anycastគឺជាការប្រាស្រ័យទាក់ទងរវាងឧបករណ៍បញ្ជូនតែមួយនិងបណ្តុំនៃ IP adder sses

Q—តើ Range នៃ IP multicat addresses មានអ្វីខ្លះ *?*

A—224.0.0.0 to 239.255.255.255.

Q—តើ IGMP មានគោលបំណងអ្វីខ្លះ *?*

A—IGMP ត្រូវបានប្រើប្រាស់រវាង Host ជាច្រើននិង local multicast router របស់វាដើម្បីចូលរួមក្នុងក្រុមនិង ចាកចេញពី multicast groups.

Q—តើអ្វីខ្លះជាគុណសម្បត្តិនៃ IGMPv2 មកលើ IGMPv1 ?

A—IGMPV2 មាន leave group message ដែលកាត់បន្ថយនូវ latency នៃចរាចរណ៍ដែលមិនត្រូវការនៅលើ LAN.

Q—តើអ្វីខ្លះជាគុណវិបត្តិនៃ *IGMP snooping* មកលើ *CGMP* ចំពោះ *low-end Layer 2 switch ?*

A—IGMP snooping ទាមទារឲ្យ switch ត្រួតពិនិត្យនៅគ្រប់ multicast packet សម្រាប់ IGMP control message ។ ចំពោះ switch វិញ វាមានផលប៉ះពាល់យ៉ាងខ្លាំងចំពោះសមត្ថភាព

Q—តើអ្វីជាគុណសម្បត្តិនៃ *shortest path (or source) trees* បើប្រៀបធៀបទៅនិង *shared trees ?*

A—Source trees ជានាផ្លូវបញ្ជូនស្រេចចិត្តរវាងឧបករណ៍បញ្ជូននិងឧបករណ៍ទទួលនីមួយៗមានន័យថាវាបាន កាត់បន្ថយនូវ ណេតវើកlatency.

Q—តើអ្វីជាគុណសម្បត្តិនៃការប្រើ *shared trees ?*

A—Shared trees ទាមទារនូវលក្ខណៈរបស់វាតិចបំផុតដែលស្ថិតនៅក្នុង routers មានន័យថាត្រូវការ Memory តិច

Q—តើព័ត៌មានអ្វីខ្លះដែល *router* ប្រើដើម្បីធ្វើ *RPF check ?*

A—The unicast routing table.

Q—ហេតុអ្វីបានជា protocol-independent multicast ត្រូវបានគេហៅថាជា"independent" ?

A—PIM ធ្វើការជាមួយ IP unicast routing protocol—RIP, EIGRP, OSPF, BGP or static routes.

Q—តើអ្វីជាគុណសម្បត្តិនៃ *MBGP ?*

A—Providers can have noncongruent unicast and multicast routing topologies.

Q—*How do RPs learn about sources from other RPs with MSDP*?

A—RPs are configured to be MSDP peers with other RPs. Each RP បញ្ជូនបន្ត source active (SA) messages to each other.

Q—*What is the purpose of the anycast RP?*

A—Load balancing and fault tolerance.

ជំពូកទី៥

Routing Algorithm

៥-១-Distance Vector Algorthm

សេចក្តីផ្តើម

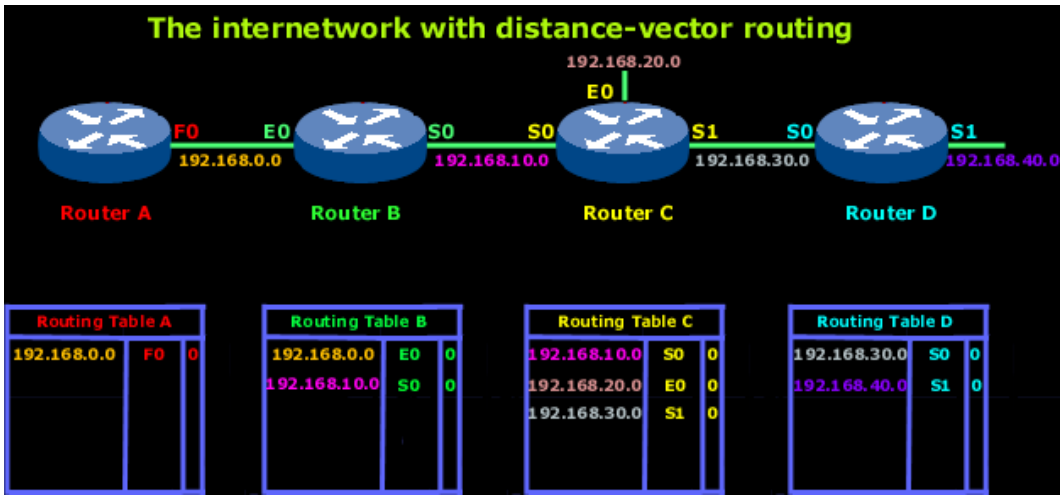
Distance Vector routing protocols ប្រើ broadcasts (255.255.255.255 ឬ FF: FF: FF: FF) របស់ routing table ទាំងមូលរបស់វារៀងរាល៣០វិនាទី។ នៅគ្រប់ interfaces របស់វាក្នុងគោលបំណងប្រាស្រ័យទាក់ទងជាមួយ Router ជិតខាងរបស់វា។ នៅពេលដែល routing tables កាន់តែធំ broadcasts ក៏មានកាន់តែច្រើនដែរ។ វិធីសាស្ត្រនេះកំណត់ទំហំរបស់ ណេតវើកលើ Distance Vector ដែលត្រូវប្រើប្រាស់ ។

Routing Information Protocol (RIP) និង Interior Gateway Routing Protocol (IGRP) គឺជា Distance Vector routing protocols ពីរដែលគេនិយមប្រើបំផុត។ អ្នកអាចស្វែងរក link ចំពោះព័ត៌មានបន្ថែមអំពី protocols ទាំងនេះ ។

Distance Vector protocols បានមើល networks សំដៅលើ routers ដែលស្ថិតនៅជាប់ៗគ្នានិង hop counts ដែលកើតឡើងហើយត្រូវបានប្រើជា metric។ "hop" count (ចំនួនអតិបរមាគឺ១៥សម្រាប់ RIP និង ១៦ គឺជា unreachable និង 255 សម្រាប់ IGMP) និងកើនម្តងមួយៗរៀងរាល់ពេលដែល Packet ឆ្លងកាត់ router ។

ដូច្នេះrouterធ្វើការសំរេចចិត្តអំពីផ្លូវដែល packet មួយនិងធ្វើដំណើរដោយពឹងផ្អែកលើចំនួន hops ដែលវាគិតដើម្បីមកដល់គោលដៅហើយបើផ្លូវពីរផ្សេងគ្នាទៅដល់គោលដៅតែមួយវានិងបញ្ជូនតាមផ្លូវដែលខ្លីជាងគេបំផុតដោយមិនគិតពីល្បឿន។ នេះគេហៅថា pin hole congestion ។

ខាងក្រោមនេះគឺជា routing table របស់ router មួយដែលប្រើ Distance Vector routing protocols:



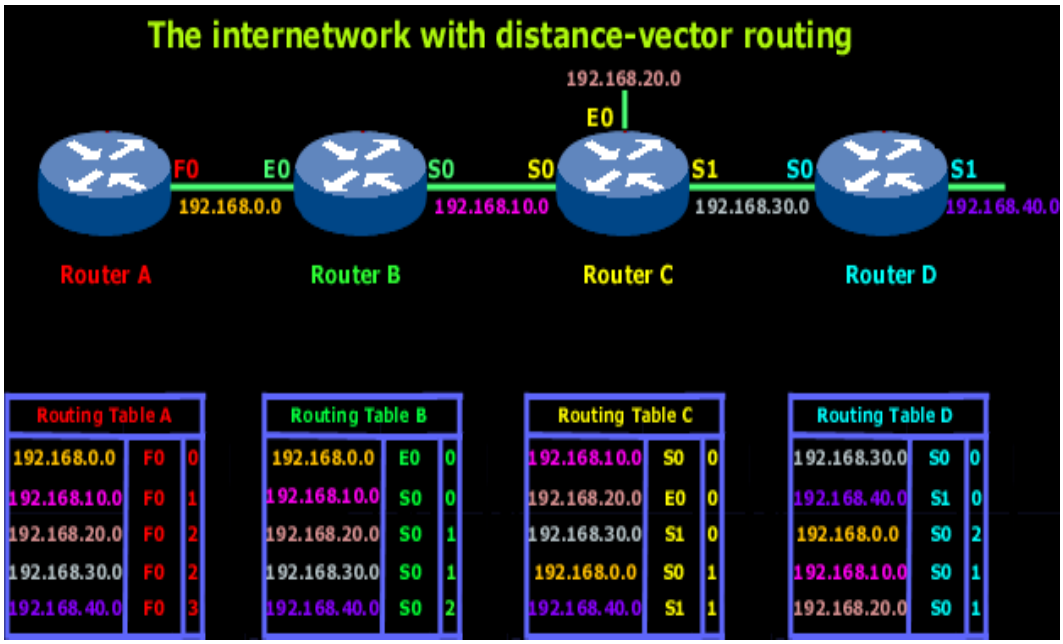
សូមស្តាប់ការពន្យល់

នៅក្នុងរូបភាពខាងលើ អ្នកឃើញ routers ចំនួន៤ ដែលនីមួយៗភ្ជាប់ជាមួយ Router ជិតខាងរបស់វាតាមប្រភេទនៃ WAN link មួយចំនួនដូចជា ISDN ។

នៅពេលបើកភ្លើងរបស់ router វានិងស្គាល់ភ្លាមអំពី networks ដែលវាអាចភ្ជាប់ជាមួយតាម interface ។ ក្នុងករណីនេះ **Router B** ស្គាល់ interface E0 បានភ្ជាប់ជាមួយ 192.168.0.0 ណេតវើកហើយ S0 interface ត្រូវបានភ្ជាប់ជាមួយ 192.168.10.0 network ។

សូមពិនិត្យម្តងទៀតនៅឯ routing table ចំពោះ Router B ចំនួនដែលអ្នកបានឃើញនៅខាងស្តាំនៃ interfaces គឺជា "hop counts" ដែលជា metric ដែល distance vector protocols ប្រើសម្រាប់រក្សាពីរបៀបរកផ្លូវបញ្ជូនចំពោះ ណេតវើកពិសេសណាមួយ។ ដោយហេតុថា networks ចំនួនពីរត្រូវបានភ្ជាប់ដោយផ្ទាល់ជាមួយ interface របស់ router វានិងមានតម្លៃសូន្យក្នុង table entry របស់ router។ ច្បាប់ដូចគ្នានេះអនុវត្តចំពោះគ្រប់ router ក្នុងឧទាហរណ៍។

យើងត្រូវចាំថា យើងគ្រាន់តែបើកភ្លើងរបស់ Router ដូច្នេះ ណេតវើកត្រូវបាន converge មានន័យថាគ្មាន data ត្រូវបញ្ជូន។ នៅពេលខ្ញុំនិយាយថា“គ្មាន data”មានន័យថា data ពីកុំព្យូទ័រណាមួយឬ server អាចនៅលើ networks ណាមួយ។ ក្នុងពេល "convergence" នេះ មានតែ data មួយប្រភេទត្រូវបានឆ្លងកាត់រវាង routers ដែលអនុញ្ញាតឲ្យវាផ្សព្វផ្សាយ routing tables របស់វានិងក្រោយមក routers និងបញ្ជូន data គ្រប់ប្រភេទទាំងអស់រវាងគ្នាវា។ ហេតុដូច្នោះហើយមានជារយៈពេល convergence គឺជាគុណវិបត្តិដ៏ធំមួយ។ បញ្ហាមួយរបស់វាជាមួយ RIP គឺថាវាមាន convergence timeយឺត ។



ចូរពន្យល់ពីអ្វីដែលយើងបានឃើញ :

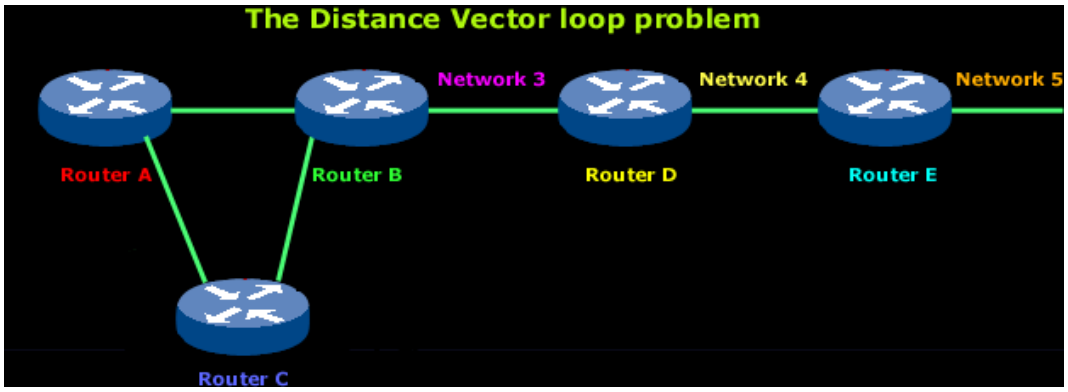
ក្នុងរូបភាពខាងលើ ណេតវើកត្រូវបានគេនិយាយថាមាន "converged" គ្រប់ routers ទាំងអស់នៅលើ ណេតវើកបានផ្សព្វផ្សាយ routing table របស់វាហើយត្រូវដឹងពី networks ដែលវាបានទាក់ទងដោយ ណេតវើកត្រូវបាន converge កុំព្យូទ័រក្នុង networks ខាងលើអាចទាក់ទងគ្នាបានជាមួយគ្នា ។

ម្តងទៀតសូមពិនិត្យមើល routing tables មួយដែលអ្នកនិងចំណាំថា network address ដែលមាន interface នៅខាងស្តាំហើយបន្ទាប់មកទៀតគឺថាវាជា hop count ចំពោះ network ។ ត្រូវចាំថា RIP និងមានតែ 15 hops ប៉ុណ្ណោះបើលើសពីនោះ packet ត្រូវបានបោះចោល ។ router នីមួយៗនិង broadcast routing table ទាំងមូលរបស់វាគ្រប់៣០វិនាទីម្តង ។

Routing ដែលពឹងផ្អែកលើ Distance Vector អាចបណ្តាលឲ្យមានបញ្ហាច្រើននៅពេល links ភ្ជាប់និងផ្តាច់ដែលជាហេតុបណ្តាលឲ្យមាន loop មិនកំណត់ហើយអាចជា de-synchronise network ។

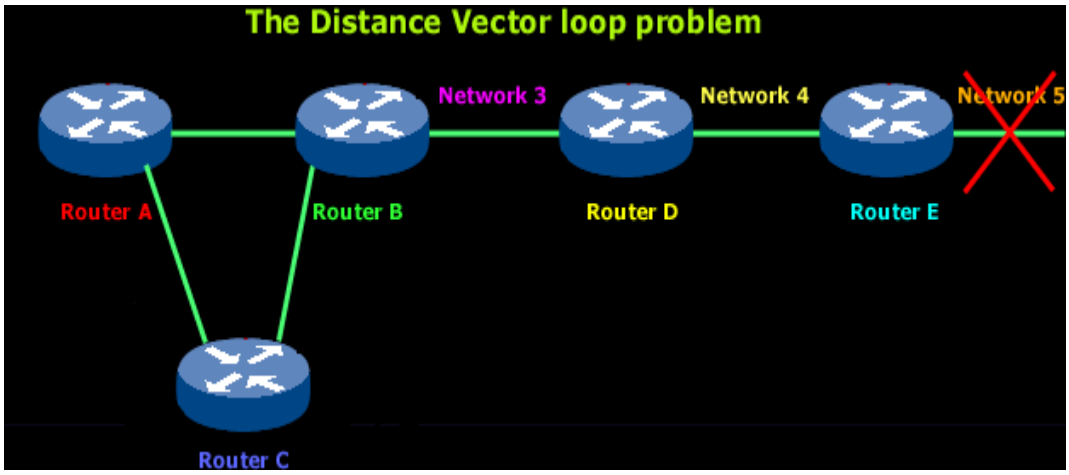
Routing loops អាចកើតឡើងនៅពេលគ្រប់ router មិនត្រូវបាន updated នៅពេលតែមួយ ។

សូមពិនិត្យមើលបញ្ហាមុនពេលយើងពិនិត្យមើលពីដំណោះស្រាយ:

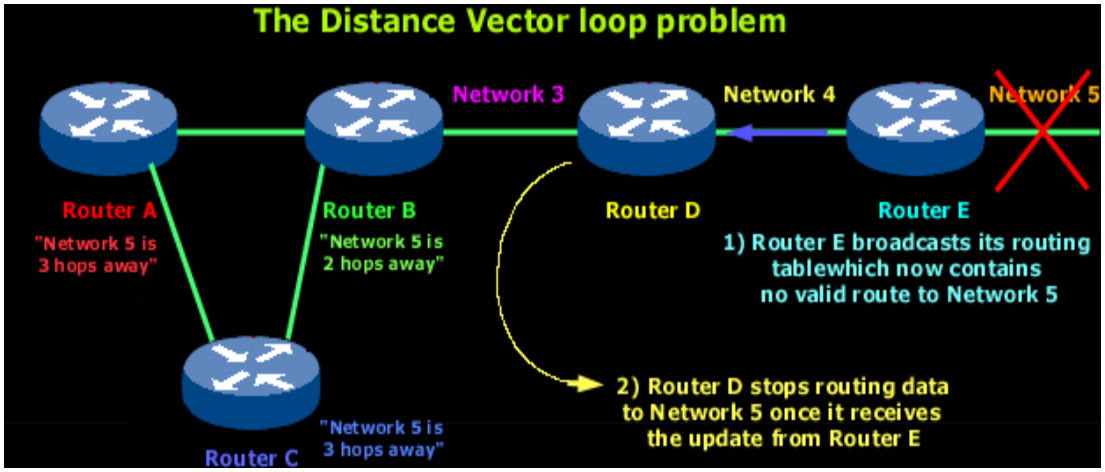


សូមពន្យល់

ក្នុងរូបភាពខាងលើអ្នកងាចឃើញ Router ៥ ដែល routers A និង B ត្រូវបានភ្ជាប់ជាមួយ Router C ហើយវាទាំងភ្ជាប់ជាមួយគ្នាតាម routers D និង E ចំពោះ ណេតវើក5 ។

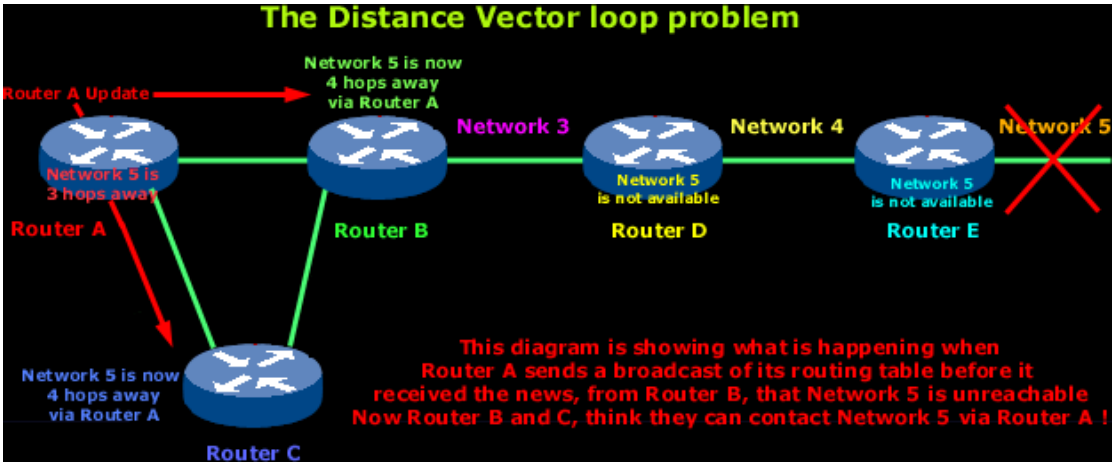


ដូចរូបភាពខាងលើ ណេតវើក5 បរាជ័យ ។



គ្រប់ routers ទាំងអស់ស្គាល់អំពី ណេតវើក5 ពី Router E ។ ឧទាហរណ៍ Router A ក្នុង tables របស់វា មានផ្លូវមកកាន់ ណេតវើក5 តាមរយៈ routers B,D និង E ។

នៅពេល ណេតវើក5 បរាជ័យ Router E បានស្គាល់វាអំពីវាពីព្រោះវាបានភ្ជាប់ជាវាដោយផ្ទាល់ហើយប្រាប់ Router D អំពីវានៅលើ update លើក្រោយរបស់វា (នៅពេលវា broadcast routing table របស់វា)។ នេះនិងផ្តល់លទ្ធផលក្នុង Router D បញ្ឈប់ routing data ចំពោះ ណេតវើក5 តាមរយៈ Router E។ ប៉ុន្តែដូចអ្នកបានឃើញក្នុងរូបភាពខាងលើ routers A B និង C មិនបានដឹងអំពី ណេតវើក5 នោះទេ ដូច្នេះវានៅតែបញ្ជូន update information ។ Router D និងបញ្ជូន update របស់វានិងធ្វើឲ្យ Router B បញ្ឈប់ routing ចំពោះ ណេតវើក5 ប៉ុន្តែ routers A និង C មិនត្រូវបាន updated ។ ចំពោះវា វានិងបង្ហាញ ណេតវើក5 មានតាមរយៈ Router B ជាមួយ metric នៃ 3 ។



ឥឡូវនេះ Router A បញ្ជូន broadcast របស់វាជាទៀងទាត់នៅ routing table របស់វាដែលរួមមានភាពដែលអាចភ្ជាប់បានសម្រាប់ ណេតវើក5។ Routers C និង B ទទួលព័ត៌មានថ្មីថា ណេតវើក5 អាចភ្ជាប់ជាមួយវាពីRouter A ដូច្នេះវាបញ្ជូនព័ត៌មានថា ណេតវើក5 មានត្រៀមជាស្រេច ។

ចាប់ពីពេលនេះទៅ packet ណាមួយមានគោលដៅជា ណេតវើក5 និងទៅ Router A ហើយបន្ទាប់មក Router B និងបានត្រឡប់មក Router A (ត្រូវចាំថា Router B ទទួលព័ត៌មានថ្មីថា ណេតវើក5 បានត្រៀមរួចរាល់តាម Router A) ។

ដូច្នេះវាគឺជាការទទួលបាន bit រញ៉េរញៃហើយអ្នកទទួលបាន loop ដ៏អស្ចារ្យមួយដែល data បានឆ្លងកាត់ពី router មួយទៅ Router មួយទៀត។ វាហាក់ដូចជាការលេងពីងប៉ុងដែរ។ ដើម្បីដោះស្រាយបញ្ហាគេប្រើបច្ចេកទេសដូចខាងក្រោម:

៥-១-១-ចំនួន Hop Count អតិបរមា

routing loop ដែលយើងទើបពិនិត្យត្រូវបានគេហៅថាជា "counting to infinity" ហើយដែលបណ្តាលដោយព័ត៌មានខុសដែលកំពុងប្រាស្រ័យទាក់ទងរវាង routers និង routers។ បើគ្មានអ្វីទប់ស្កាត់ប្រភេទៃ loop នេះទេ hop count និងរក្សាការកើនឡើងគ្រប់ពេលដែល Packet ឆ្លងកាត់ router មួយ។ វិធីមួយនៃការដោះស្រាយបញ្ហានេះគឺកំណត់ចំនួន hop count អតិបរមា។ Distance Vector(RIP) អនុញ្ញាតឲ្យ hop count មានតម្លៃរហូតដល់ 15 ដូច្នេះបើលើសពី១៥(១៦)គឺជា Network ដែលមិនអាចភ្ជាប់បាន។ ដូច្នេះបើមាន loop មួយកើតឡើង វានិងធ្វើដំណើរជុំវិញ ណេតវើករហូតទាល់តែ packet មានតម្លៃនៃ hop count ស្មើ 15 ហើយ next router និងបោះ packet ចោល ។

៥-១-២-Split Horizon

ការធ្វើការលើគោលការណ៍ដែលវាគ្មានសារៈសំខាន់នោះគឺការបញ្ជូនព័ត៌មានអំពី router មួយត្រឡប់ឲ្យគោលដៅ packetដើមវិញ។ ដូច្នេះបើខ្ញុំប្រាប់អ្នកអំពីរឿងកំប្លែងមួយវាមិនអាចនិយាយថាអ្នកប្រាប់ខ្ញុំវិញអំពីរឿងកំប្លែងនោះម្តងទៀតទេ ។

នៅក្នុងឧទាហរណ៍របស់យើង មាន Router A មួយត្រូវបានគេការពារពីការបញ្ជូនព័ត៌មានត្រូវបាន update ដែលវាបានទទួលពី Router B ត្រឡប់មក Router B វិញនោះបានទេ។

៥-១-៣-Route Poisoning

គឺជាដំណោះស្រាយមួយទៀតចំពោះ split horizon។ នៅពេលដែល router មួយទទួលព័ត៌មានអំពី route មួយពី ណេតវើកពិសេសណាមួយ router ក៏ផ្សព្វផ្សាយអំពី route ត្រឡប់ឲ្យ ណេតវើកនោះជាមួយ metric ស្មើ 16 ដែលចង្អុលបង្ហាញថាគោលដៅ ណេតវើកមិនអាចភ្ជាប់បាន ។

ក្នុងឧទាហរណ៍របស់យើងគឺមានន័យថានៅពេល ណេតវើក5 ខូច Router E ក៏ចាប់ផ្តើម router ដែលកំពុងបំពុលដោយការបញ្ជូលនៅ table entry មួយសម្រាប់ ណេតវើក5 ជាតម្លៃ 16 មានន័យថាមិនអាចភ្ជាប់បាន (unreachable) ។ ក្នុងវិធីនេះ Router D មិនទទួលការ updates ខុសអំពី route ចំពោះ ណេតវើក5។ នៅពេល

Router D ទទួល router មួយដែលកំពុងបំពុលពីRouter E វាបញ្ជូន update មួយហៅថា poison reverse ត្រឡប់មក Router E ។ នេះធ្វើឲ្យគ្រប់ផ្លូវបញ្ជូនទាំងអស់នៅលើ segment បានទទួលព័ត៌មានដែលបានបំពុល ។

ផ្លូវបញ្ជូនដែលបានបំពុលបានប្រើជាមួយ hold-downs និងបង្កើនល្បឿននៅរយៈពេល convergence ពីព្រោះrouters ដែលនៅជិតៗវាគ្មានរង់ចាំរហូតដល់ ៣០វិនាទីមុនពេលផ្សព្វផ្សាយផ្លូវបញ្ជូនដែលបានបំពុល ។

៥-១-៤-Hold-Down Timers

Routers រក្សាទុក entry មួយដែលនិយាយអំពី ណេតវើកបានខូចដែលកំពុងអនុញ្ញាតឲ្យ routers ផ្សេងៗទៀតគណនាអំពី topology ដែលបានប្តូរ។ តាមវិធីនេះការអនុញ្ញាតពេលវេលាចំពោះ router ដែលខូចត្រឡប់មកដើមវិញឬ ណេតវើកដែលមានស្ថេរភាពមុនពេលវាប្តូរចំពោះផ្លូវបញ្ជូនបន្ទាប់ ។

នៅពេល router មួយទទួលការ update ពីអ្នកជិតខាងវាបង្ហាញថា accessible ណេតវើកពីមុនមិនដំណើរការហើយមិនអាច access បាន។ hold-down timer និងដំណើរការ។ បើ update ថ្មីមួយទទួលពី neighbor មួយមាន metric មួយល្អប្រសើរជាង ណេតវើកentry ដើមរបស់វា hold-down ត្រូវបានលុបចោលហើយ data ត្រូវបានបញ្ជូន។ ប៉ុន្តែ update មួយត្រូវបានទទួលពី neighbor router មួយមុនពេល hold-down timer ត្រូវបានផុតកំណត់និងមាន metric តិចជាង route ពីមុន។ ដូច្នេះ update មិនត្រូវបានអើពើហើយ hold-down timer នៅតែមាន។ វាអនុញ្ញាតឲ្យមានពេលវេលាច្រើនសម្រាប់ ណេតវើកចំពោះ converge ។

Hold-down timers ប្រើ triggered updates ដែលធ្វើឡើងវិញចំពោះ hold-down timer គឺជាការប្រកាសឲ្យដឹងពី Router របស់អ្នកជិតខាងដែលបានប្តូរក្នុង network។ មិនដូចទៅនិង update messages ពី routers ជិតខាង triggered updates បង្កើត routing table ថ្មីដែលត្រូវបានបញ្ជូនភ្លាមឲ្យ neighbor routers ពីព្រោះវាបានដឹងថាមានការផ្លាស់ប្តូរក្នុង ណេតវើក។

មាន instances បីកើតឡើងនៅពេល triggered updates និងធ្វើឡើងវិញចំពោះ hold-down timer:

1) hold-down timer ត្រូវបានផុតកំណត់

2) router បានទទួល processing task ជាផ្នែកៗចំពោះចំនួនរបស់ links ក្នុង internetwork

3) update មួយផ្សេងទៀតត្រូវបានទទួលចង្អុលបង្ហាញថា ណេតវើកត្រូវបានផ្លាស់ប្តូរ

នៅក្នុងឧទាហរណ៍របស់យើង update ណាមួយទទួលបានដោយ Router B ពី Router A និងមិនទទួលលុះត្រាតែ hold-down timer ត្រូវបានផុតកំណត់។ នេះប្រាកដថា Router B និងមិនទទួលការ update ខុសពី routers ណាមួយដែលមិនបានស្គាល់ថា ណេតវើក5 មិនអាចភ្ជាប់បាន។ បន្ទាប់មក Router B និងបញ្ជូន update មួយហើយបានកែ tables របស់ routers ផ្សេងទៀត ។

៥-២-សេចក្តីផ្តើមចំពោះ Link State

Link State protocols មិនដូចទៅនិងDistance Vectorដែល broadcasts របស់វាប្រើ multicast ។

Multicast គឺជា "broadcast" មួយចំពោះក្រុមនៃ hosts ក្នុងករណី Router នេះ បើយើងមាន Routers ចំនួន១០ដែល Router ចំនួន៤គឺជាផ្នែកមួយនៃ "mutilcast group" មួយនោះនៅពេលយើងបញ្ជូនចេញ multicast packet មួយឲ្យក្រុមនេះគឺមានតែ Routers ចំនួន៤ប៉ុណ្ណោះដែលទទួល updates រីឯផ្នែកនៅសល់វា និងមិនអើពើចំពោះdata ។ multicast address គឺមានលេខជា 224.0.0.5 & 224.0.0.6 ដែល address នេះ កំណត់ដោយ IGRP (Interior Gateway Routing Protocol) ។

Link State routing protocols មិនបានមើលពី routers ដែលនៅជាប់ៗគ្នានិង hop counts នោះទេ ប៉ុន្តែវាបង្កើតនៅការពិនិត្យមើល ណេតវើកទាំងមូលដែលពិពណ៌នាអំពីផ្លូវបញ្ជូនដែលអាចបញ្ជូនបានជាមួយតម្លៃ របស់វា ។ ការប្រើ SPF (Shortest Path First) algorithm router បង្កើតជា "topological database" ដែលជា ទម្រង់នៃរចនារបស់ ណេតវើកrouters ដែលវាបានដឹងពីខ្លួនវា ។ បន្ទាប់មកវាជាក់ខ្លួនវានៅផ្នែកខាងលើបំផុតនៃ ទម្រង់រៀបតាមលំដាប់ ។

នៅពេល router មួយប្រើ Link State protocol មួយដូចជា OSPF (Open Shortest Path First) ដែល ស្គាល់អំពីការប្តូរលើ ណេតវើកវានិង multicast ការផ្លាស់ប្តូរនេះ ដូចនេះវា flood ឲ្យ ណេតវើកជាមួយព័ត៌មាននេះ ។ ព័ត៌មានដែលroutersត្រូវការសម្រាប់បង្កើតdatabasesរបស់វាត្រូវបានផ្តល់ក្នុងទម្រង់របស់ Link State advertisement packets (LSAP) ។ Routers មិនផ្សព្វផ្សាយនៅ routing tables ទាំងមូលនោះទេ ផ្ទុយទៅវិញ router នីមួយៗផ្សព្វផ្សាយតែ ព័ត៌មានរបស់វាប៉ុណ្ណោះដោយគិតពី routers ដែលស្ថិតនៅជាប់គ្នា ។

Link State protocols ប្រៀបធៀបជាមួយ Distance Vector protocols មានដូចខាង ក្រោម:

- ត្រូវការទំហំ Memory ធំ
- ការគណនា Shortest path ត្រូវការ CPU circles ជាច្រើន
- បើ ណេតវើកមានស្ថេរភាពហើយមានការប្រើប្រាស់ bandwidth តិចនោះវាមានប្រតិកម្មរហ័សចំពោះ ការផ្លាស់ប្តូរ topology
- ការផ្សព្វផ្សាយមិនអាចច្រោះបានទេ ។ គ្រប់ Items ទាំងអស់នៅក្នុង database ត្រូវតែបញ្ជូនឲ្យ Router នៅ ជិតខាង
- គ្រប់Router នៅជិតខាងត្រូវតែមានទំនុកចិត្ត
- Authentication mechanisms ត្រូវបានប្រើប្រាស់ដើម្បីជៀសវាងភាពជាប់គ្នាដែលមិនត្រូវការ
- គ្មាន split horizon កើតឡើង

ទោះបីជា Link State protocols ដំណើរការបានល្អយ៉ាងណាក៏ដោយបញ្ហានៅតែកើតមានឡើង។ តាមធម្មតាបញ្ហាអាចកើតឡើងបណ្តាលមកពីមានការផ្លាស់ប្តូរក្នុង ណោតរើកtopology(ដូចជាមានខ្សែដាច់ជាដើម) ហើយគ្រប់ Routers មិនទទួលបានការ Update ភ្លាមនោះទេពីព្រោះវាស្ថិតនៅលើកំរិតល្បឿនខុសគ្នា ដូច្នេះ Router ដែលភ្ជាប់តាមបណ្តាញដែលមានល្បឿនលឿននិងទទួលបានការ Update លឿនជាង។

បច្ចេកទេសផ្សេងៗត្រូវបានបង្កើតឡើងដើម្បីដោះស្រាយបញ្ហាមានដូចជា

1) កាត់បន្ថយការ update ជាញឹកញាប់

2) Target link-state updates ប្រើ multicast

3) ប្រើ link-state area hierarchy សម្រាប់ topology

4) ផ្លាស់ប្តូរព័ត៌មានសង្ខេបពីផ្លូវបញ្ជូននៅឯ area borders

5) ប្រើប្រាស់ Time-stamps Update numbering & counters

6) គ្រប់គ្រង់ partitions ដោយប្រើ area hierarchy

ជំពូកទី៦

មូលដ្ឋានគ្រឹះនៃ Networking

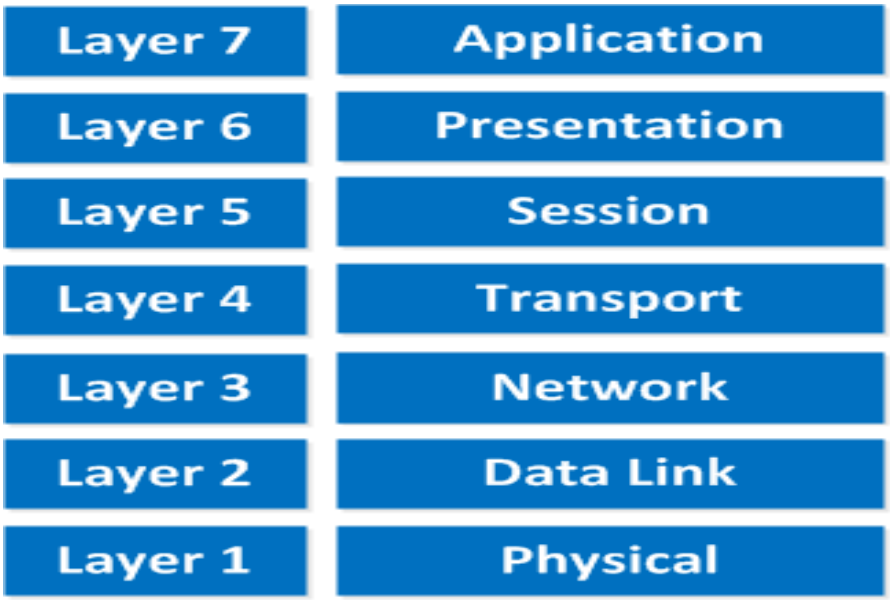
៦-១-សេចក្តីផ្តើមចំពោះ OSI Model

នៅពេលចាប់ផ្តើមបង្កើតនូវប្រព័ន្ធ ណេតវើកវាស្ថិតនៅក្នុងភាពសាំញ៉ាំ។ ក្រុមហ៊ុននីមួយៗមានដំណោះស្រាយផ្ទាល់ខ្លួន។ ផ្នែកមួយដែលលំបាកគឺដំណោះស្រាយរបស់ពួកគេមិនត្រូវគ្នាឬមិនចុះសំរុងជាមួយគ្នា។ ត្រង់ចំណុចនេះហើយដែលឈានទៅរកការបង្កើតបានជា ISO Model នេះឡើងដែលមានស្រទាប់សម្រាប់បម្រើទៅតាម hardware របស់អ្នកផលិតវាសម្រាប់ប្រើប្រាស់ជាមួយ ណេតវើកហើយក្រុមហ៊ុនផ្សេងទៀតអាចផលិត Software សម្រាប់ application layer។ ការប្រើគំរូបើកទូលាយដែលអ្នកគ្រប់គ្នាយល់ព្រមទាំងអស់គ្នា មានន័យថាយើងអាចកសាងនូវ ណេតវើកដែលចុះសំរុងជាមួយគ្នា។

ដើម្បីដោះស្រាយបញ្ហានេះបាន ISO បានធ្វើការស្រាវជ្រាវទៅលើគំរូនៃ ណេតវើកប្រភេទផ្សេងគ្នាហើយលទ្ធផលនៃការសិក្សានោះគឺបង្កើតបានជា OSI Model មួយដែលបានចេញឲ្យប្រើប្រាស់នៅឆ្នាំ១៩៨៤។ សព្វថ្ងៃនេះមានក្រុមហ៊ុនភាគច្រើនបង្កើត ណេតវើកដោយពឹងផ្អែកទៅលើ OSI Model នឹងប្រើប្រាស់នៅប្រភេទ hardware ផ្សេងៗគ្នាអាចប្រើជាមួយគ្នាបាន។

OSI Model មិនគ្រាន់តែជាគំរូមួយសម្រាប់ឲ្យ ណេតវើកអាចធ្វើការជាមួយគ្នាបាន វាក៏ជាវិធីមួយដ៏ល្អប្រសើរបំផុតសម្រាប់បង្រៀនមនុស្សឲ្យងាយយល់ពី ណេតវើក។

ខាងក្រោមនេះគឺជា OSI Model ដែលមាន ៧ស្រទាប់គឺ:



នេះគឺជា OSI Model ដែលមាន៧ស្រទាប់ ។ យើងនឹងសិក្សាពីស្រទាប់ខាងក្រោមបំផុតទៅដល់ខាងលើបំផុត។

Physical Layer: ស្រទាប់នេះគឺពិពណ៌នាអំពីកម្រិតនៃវ៉ុលរបស់ចរន្ត ពីការកំនត់ពេលវេលា ពីអត្រានៃការបញ្ជូនទិន្នន័យនិងពីការតភ្ជាប់ជាដើម។ អ្វីៗដែលអ្នកអាចប៉ះបានគឺជា Physical Layer ។

Data Link: នៅក្នុងស្រទាប់នេះគឺបានកំណត់ពីរបៀបរៀបចំទិន្នន័យ កែទៅលើ Error នឹងធានាថាទិន្នន័យត្រូវបានចែកចាយត្រឹមត្រូវ។ នៅក្នុងស្រទាប់នេះគឺមាន MAC Addresses នឹង Ethernet frames ដែលស្ថិតនៅក្នុង Data Link layer។

Network: ស្រទាប់នេះគឺមានតួនាទីសម្រាប់ភ្ជាប់នឹងជ្រើសរើសយកផ្លូវបញ្ជូន (Routing)។ នេះក្នុងស្រទាប់នេះគឺជាកន្លែងដែល IPv4 និង IPv6 ស្ថិតនៅ។ គ្រប់ ណេតវើកdevice ត្រូវការនូវ Address តែមួយគត់នៅលើ network។

Transport: មានតួនាទីទទួលខុសត្រូវក្នុងការបញ្ជូនទិន្នន័យនៅពេលដែលអ្នក Download អត្ថបទពី Internet ។ នៅក្នុងស្រទាប់នេះមាន Protocols ដូចជា:

TCP:គឺជា Protocol ដែលទទួលខុសត្រូវក្នុងការបញ្ជូនទិន្នន័យ

UDP:មិនទទួលខុសត្រូវក្នុងការបញ្ជូនទិន្នន័យ

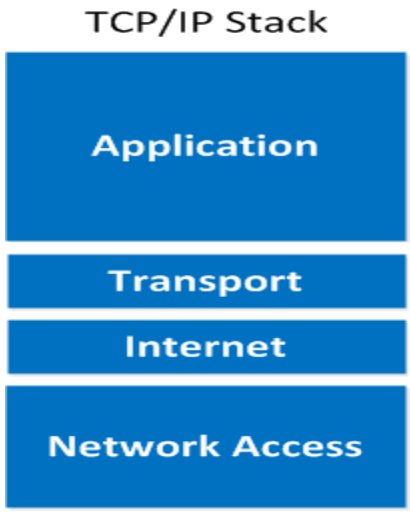
Session: ទទួលខុសត្រូវក្នុងការបង្កើតការភ្ជាប់ គ្រប់គ្រងការភ្ជាប់និងផ្តាច់ការភ្ជាប់រវាង Host ពីរ។ នៅពេលអ្នក browse ចំពោះ website មួយនៅលើ internet Webservers កំពុងត្រូវការកត់ត្រានូវគ្រប់ “sessions” ទាំងអស់ផ្សេងៗគ្នា។

Presentation: វាមានតួនាទីធ្វើឲ្យព័ត៌មានអាចអានបានដោយ application layer។ កុំព្យូទ័រភាគច្រើនប្រ ASCII កូដ។ បើកុំព្យូទ័រផ្សេងប្រើកូដផ្សេងដូចជា EBCDIC បន្ទាប់មកវានឹងរៀបចំឡើងវិញដើម្បីឲ្យកុំព្យូទ័រទាំងពីរអាចយល់គ្នាបាន។

Application: នេះគឺជា applications រួមមានដូចជា E-mail, browse web (HTTP), FTP ។ល។

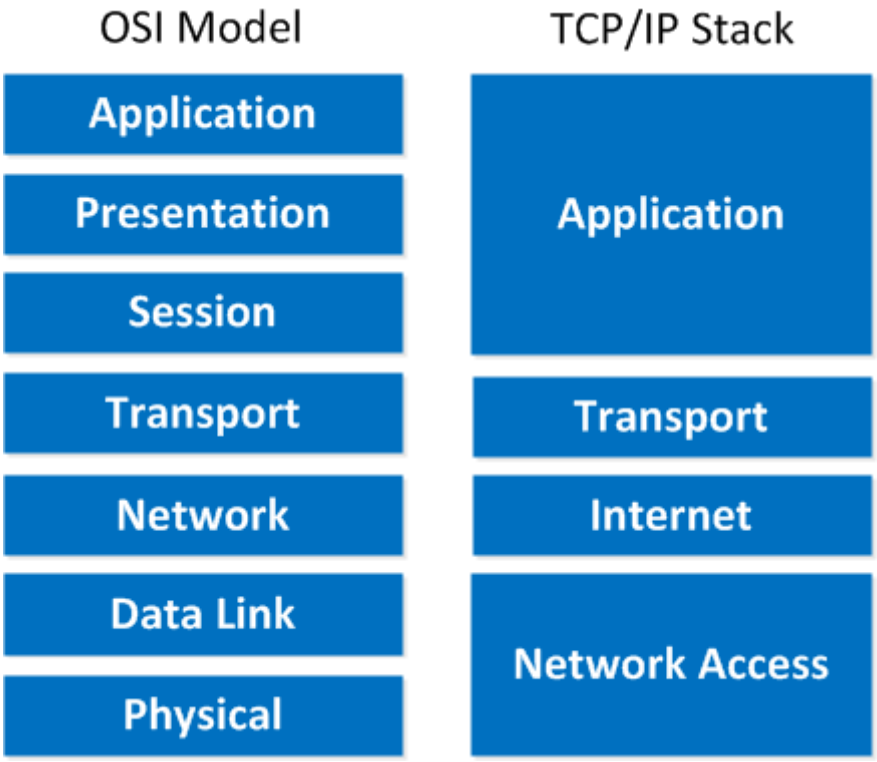
៦-២-TCP/IP Stack

ក្រៅពី OSI Model នៅមានការរៀបចំមួយប្រភេទទៀតដែលមានគំរូប្រហាក់ប្រហែលគ្នាក៏មានប្រជាការនិយមដែរ។ Model នោះគឺ TCP/IP stack ដូចបង្ហាញខាងក្រោម:



ដូចអ្នកបានឃើញខាងលើហើយ ៣ស្រទាប់ខាងលើនៃ OSI Model ត្រូវបញ្ចូលគ្នាជា “Application layer”។ ចំពោះ network layer ត្រូវបានគេហៅថា “Internet” layer ហើយ Layers ទាំងពីរខាងក្រោមត្រូវបានគេបញ្ចូលជា “ណេតវើកAccess” layer។

គំនូសតាងខាងក្រោមនេះគឺជាការប្រៀបធៀបរវាង Model ទាំងពីរ។



ជាមូលដ្ឋានគ្រឹះមានដូចតែគ្នា លើកលែងតែវាមានស្រទាប់ខ្លះត្រូវបានបញ្ចូលគ្នាហើយដាក់ឈ្មោះខុសគ្នាប៉ុណ្ណោះ។ physical និង data link layer ត្រូវបានគេបញ្ចូលគ្នាជា ណេតវើកaccess layer។ network layer មានឈ្មោះថាជា internet layer ហើយ session, presentation និង application layer ត្រូវបានបញ្ចូលជា application layer តែមួយ។

៦-៣-IP (Internet Protocol) Version 4

IP ប្រើ Packets ដែលមានឈ្មោះថា IP packets ដើម្បីនាំយកនូវព័ត៌មាន។ IP packet នីមួយៗគឺជាឯកតានៃព័ត៌មានតែមួយគត់ហើយក្រៅពីទិន្នន័យវានាំយកព័ត៌មានដើម្បីកំនត់ពីទីកន្លែងដែលត្រូវបញ្ជូន Packet ទៅ។ IP កំនត់ពីទីកន្លែងដែលត្រូវបញ្ជូន Packets ទៅដោយពិនិត្យទៅលើ IP address ដែលជាគោលដៅ។

សូមពិនិត្យមើលពីលក្ខណៈពិសេសរបស់វា:

ដំណើរការនៅឯស្រទាប់ network layer នៃ OSI Model

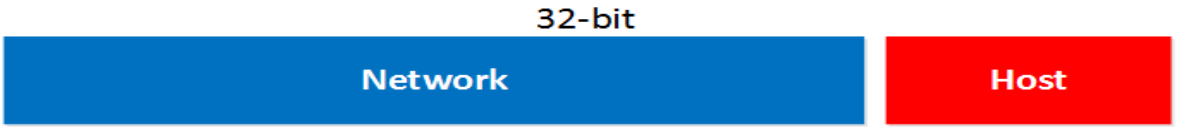
Connectionless protocol: IP ខ្លួនវាផ្ទាល់មិនបង្កើតនូវការភ្ជាប់នោះទេ។ ក្នុងគោលបំណងនៃការដឹកជញ្ជូនទិន្នន័យអ្នកត្រូវការ Transport layer ហើយប្រើ TCP ឬ UDP ។

Packet នីមួយៗត្រូវបានចាត់ទុកថាឯករាជ្យ។ ការមកដល់គោលដៅរបស់វាមិនតាមលំដាប់នោះទេ។

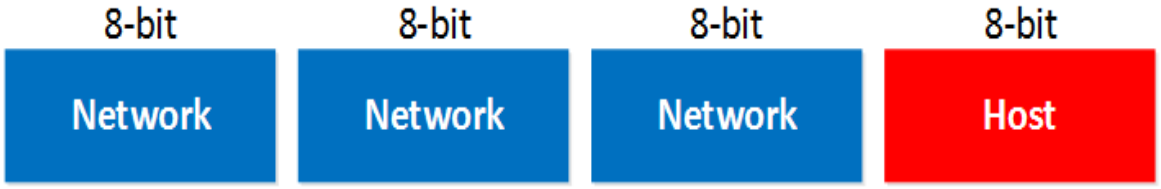
Hierarchical: IP addresses មានការរៀបចំតាមលំដាប់ថ្នាក់។

នៅពេលអ្នកត្រូវការនូវ IP address មួយដើម្បីសម្គាល់ពីឧបករណ៍របស់ ណេតវើកនីមួយៗនៅលើ ណេតវើក។ IP address មួយហាក់បីដូចជាលេខទូរស័ព្ទមួយ។ មនុស្សគ្រប់ៗគ្នានៅក្នុងទីក្រុងមួយមានទូរស័ព្ទមួយនៅឯផ្ទះហើយមានលេខទូរស័ព្ទតែមួយគត់ដែលអ្នកអាចទាក់ទងជាមួយពួកគេ។

IP address មួយមានប្រវែង 32 bits ហើយមាន២ផ្នែកគឺផ្នែកតំណាងឲ្យ ណេតវើកនិងផ្នែកតំណាងឲ្យ Host។



IP address មានប្រវែង 32 bits ប៉ុន្តែយើងសរសេរវាជា៤ផ្នែកដែលផ្នែកនីមួយៗមាន ៨ bits ។ ៨ bits ដែលគេហៅថា “byte” ។ ដូច្នេះ IP address មានទម្រង់ដូចខាងក្រោម:



ផ្នែកណេតវើកនឹងប្រាប់ឲ្យយើងដឹងពី ណេតវើកមួយណាដែល IP address ស្ថិតនៅជាមួយ។ អ្នកអាចប្រៀបធៀបវាទៅនិងទីក្រុងឬលេខកូដតំបន់នៃលេខទូរស័ព្ទ។ ចំណែកឯ Host វិញគឺជាផ្នែកតែមួយគត់ដែលសម្គាល់ពីឧបករណ៍ ណេតវើកដែលវាដូចទៅនិងលេខរបស់ទូរស័ព្ទ។

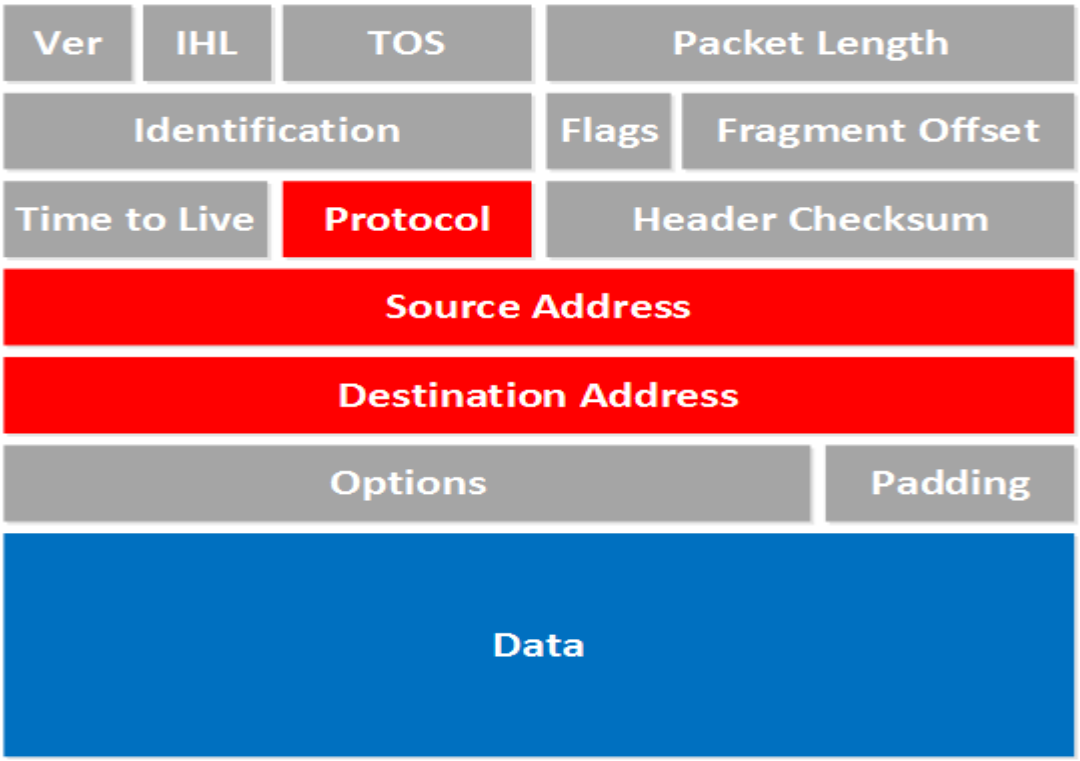
អ្នកប្រហែលជាបានឃើញ IP address 192.168.1.1 ពីមុនមកហើយ។ វាជា IP address ដែលគេនិយមប្រើប្រាស់វាជាញឹកញាប់នៅលើ local networks។ ចំពោះ IP address មាន 3 bytes ដំបូងជា “Network” ហើយ byte ចុងក្រោយជា “Host” address ។

ហេតុអ្វីបានជា 3 bytes ដំបូងជា “Network” ហើយ byte ចុងក្រោយជា “ Host” ?

ចំលើយគឺដោយសារតែ default subnetmask គឺ255.255.255.0 ។

Subnet mask គឺប្រាប់កុំព្យូទ័រឲ្យដឹងពីផ្នែកណាជា “Network” និងផ្នែកណាជា “Host” ។

សូមពិនិត្យមើលពី IP Packet ពិតប្រាកដដូចខាងក្រោម:



មានផ្នែកជាច្រើននៅទីនោះ។ សូមពិនិត្យមើលទៅលើផ្នែកមួយចំនួនដែលមានសារៈសំខាន់សម្រាប់អ្នក។

Protocol: ត្រង់នេះនិងបង្ហាញថាតើ Protocol មួយណាដែលអ្នកកំពុងប្រើនៅលើ IP។ កន្លែងនេះហើយដែលកំណត់ពី transport layer protocol មួយណាដែលយើងកំពុងប្រើវា។ វាអាចជា TCP, UDP ឬអាចជាអ្វីផ្សេង។

Source Address: អ្នកនឹងឃើញ IP address របស់ឧបករណ៍ដែលបានបង្កើត IP Packet នេះ

Destination Address: វាជា IP address របស់ឧបករណ៍ដែលត្រូវទទួល IP packet។

Data: នេះគឺទិន្នន័យពិតប្រាកដដែលទទួលបានពីឧបករណ៍ផ្សេង។

បើអ្នកធ្លាប់ប្រើ Wireshark អ្នកអាចដឹងពីផ្នែកទាំងនេះ។ នេះគឺជាការថតចេញពី IP packet នៅក្នុង Wireshark:

```
▼ Internet Protocol, Src: 192.168.69.2 (192.168.69.2), Dst: 192.168.69.1 (192.168.69.1)
    Version: 4
    Header length: 20 bytes
  ▶ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 497
    Identification: 0xf5db (62939)
  ▶ Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 64
    Protocol: TCP (6)
  ▶ Header checksum: 0x37d7 [correct]
    Source: 192.168.69.2 (192.168.69.2)
    Destination: 192.168.69.1 (192.168.69.1)
```

សូមពិនិត្យមើល IP address ខាងក្រោមនេះ

192.168.1.1

តើយើងបានដឹងអ្វីខ្លះនៅក្នុង IP address នេះ? ជាបឋមយើងបានដឹងថាវាមានប្រវែង 32 bits ដែលសរសេរនៅក្នុងប្រព័ន្ធគោល២មានទម្រង់ជា:

11000000101010000000000100000001

វាជាទម្រង់មួយដែលមនុស្សមិនអាចអានបានយ៉ាងងាយនោះទេ។ ដូច្នេះត្រូវសរសេរវាជាក្រុមដែលមាន 8 bits ដែលគេហៅថា "byte" ឬជា Octet ។

11000000 **10101000** **00000001** **00000001**

យើងក៏អាចបំប្លែងវាមកជាលេខប្រព័ន្ធគោល១០បានដែរ។ សូមពិនិត្យមើលការបំប្លែងចំពោះតំបន់ទី១ពីលេខគោល២មកជាគោល១០ដោយប្រើតារាងដូចខាងក្រោម:

Bits	128	64	32	16	8	4	2	1
	0	0	0	0	0	0	0	0

Byte ទី១

11000000

Bits	128	64	32	16	8	4	2	1
	1	1	0	0	0	0	0	0

128 + 64 = 192

Byte ទី២

10101000

Bits	128	64	32	16	8	4	2	1
	1	0	1	0	1	0	0	0

128 + 32 + 8 = 168

Byte ទី៣

00000001

Bits	128	64	32	16	8	4	2	1
	0	0	0	0	0	0	0	1

មានតែ bit ចុងក្រោយទេដែលជាលេខ១

Byte ទី៤

00000001

Bits	128	64	32	16	8	4	2	1
	0	0	0	0	0	0	0	1

ដូចគ្នាទៅនិង byte ទី៣ មានលេខគោល១០គឺ១។ ដូច្នេះយើងទទួលបាន IP address

192 **168** **1** **1**

យើងមាន class បីប្រភេទដែលយើងកំពុងប្រើវាគឺ:

Class A

Class B

Class C

តើវាទាំងអស់នេះខុសគ្នាដូចម្តេច ? ភាពខុសគ្នាគឺចំនួន Hosts ដែលវាមាននៅក្នុង ណេតវើកនីមួយៗ។

ចំពោះ 3 octets ដំបូងដែលមានពណ៌ខៀវគឺជា ណេតវើកនៃ IP address នេះ។ ចំណែកឯពណ៌ក្រហមគឺជា Hosts ។ ដូច្នេះយើងអាចនិយាយថា byte ឬ octet ចុងក្រោយគឺជា Host ។

ចំពោះ IP addresses ខាងក្រោមគឺស្ថិតនៅក្នុង ណេតវើកតែមួយ:

192.168.1.1

192.168.1.2

192.168.1.3

ពីព្រោះថាវាមាន ណេតវើកដូចគ្នា។

ចំពោះកុំព្យូទ័រដែលមាន IP address 192.168.2.1 មិនស្ថិតនៅក្នុង ណេតវើកជាមួយគ្នានោះទេ។ ពីព្រោះថាវាមានផ្នែក ណេតវើករបស់វាគឺ 192.168.2.x បើប្រៀបធៀបជាមួយ 192.168.1.x ។

តើអ្នកគិតយ៉ាងដូចម្តេចបើកុំព្យូទ័រមួយចង់បញ្ជូន IP packets ទៅឲ្យកុំព្យូទ័រមួយទៀតនៅលើ ណេតវើកផ្សេង ?

បើអ្នកកំពុងប្រើ Windows អ្នកគ្រាន់តែបើកនូវ CMD ហើយប្រើបញ្ជា ipconfig ដើម្បីមើលព័ត៌មានទាក់ទងជាមួយ IP address ។

C:Documents and Settingsកុំព្យូទ័រ>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix . :

IP Address. . . . . : 192.168.1.1

Subnet Mask . . . . . : 255.255.255.0

Default Gateway . . . . . : 192.168.1.254

កុំព្យូទ័រខាងលើស្ថិតនៅក្នុង ណេតវើក192.168.1.x។ នៅពេលវាចង់បញ្ជូនអ្វីមួយទៅឲ្យកុំព្យូទ័រមួយទៀតនៅក្នុងណេតវើកមួយផ្សេងទៀត វានឹងប្រើ default gateways របស់វា។ វាគឺជា Router។ នៅក្នុងឧទាហរណ៍ខាងលើគឺ Router មាន IP address 192.168.1.254 ។

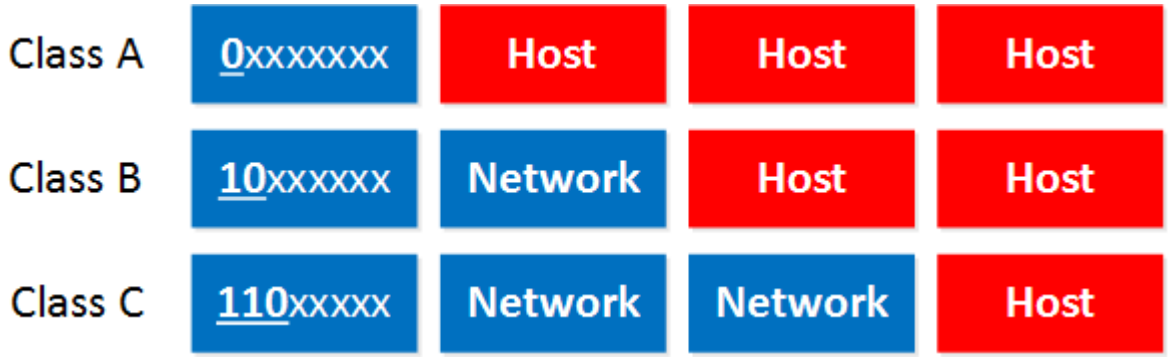
សូមពិនិត្យមើលពី Class របស់ IP addresses ខាងក្រោម:

បើអ្នកប្រើ ណេតវើកនៅក្នុង Class A អ្នកមានចំនួន Hosts ជាច្រើននៅក្នុង ណេតវើកនីមួយៗ។

បើអ្នកប្រើ ណេតវើកនៅក្នុង Class B អ្នកអាចបង្កើតបាន Networks ជាច្រើន ប៉ុន្តែមានចំនួន Hosts ក្នុងមួយ ណេតវើកចំនួនតិច។

ចំណែកឯ ណេតវើកនៅក្នុង Class C វិញអាចបង្កើតបាន ណេតវើកជាច្រើន ប៉ុន្តែមានចំនួន Hosts តិចក្នុងមួយ Network។ យើងទើបតែបានពន្យល់ពី Class C ណេតវើកដែលមាន IP address 192.168.1.1 ។

តារាងសង្ខេបពី Class A, B និង C



ចំពោះ Class A មាន bit 0 នៅក្នុង byte ទី១ជានិច្ច។

ចំពោះ Class B មាន 2 bits ដំបូងគឺជា 10 ជានិច្ចនៅក្នុង Byte ដំបូង

ចំណែកឯ Class C វិញមាន 3 bits ដំបូងជា 110 ជានិច្ចនៅក្នុង Byte ដំបូង

៦-៤-ការណែនាំឲ្យស្គាល់ចំពោះ TCP និង UDP

នៅកផ្នែកនេះយើងនឹងសិក្សាទៅលើ Transport protocols គឺ TCP និង UDP។ បើអ្នកបានដឹងពី IP និង IP Packets អ្នកដឹងថាវាត្រូវការនូវ Transport protocol ដើម្បីបញ្ជូន IP Packets។ Transport Protocols ដែលគេនិយមប្រើគឺ

TCP (Transmission Control Protocol)

UDP (User Datagram Protocol)

ហេតុអ្វីបានជាយើងលើកយកតែ Transport protocols ពីរ? ហើយអ្វីបានជាយើងចាប់អារម្មណ៍តែទៅលើវាដោយមិនបានចាប់អារម្មណ៍ទៅលើ Protocol ផ្សេងទៀត?

ចំលើយដ៏ខ្លីគឺ

TCP គឺជា protocol ដែលទទួលខុសត្រូវក្នុងការបញ្ជូនទិន្នន័យ

UDP គឺជា Protocol ដែលមិនទទួលខុសត្រូវក្នុងការបញ្ជូនទិន្នន័យឬជា best-effort protocol.

តើ Protocol មិនទទួលខុសត្រូវក្នុងការបញ្ជូនទិន្នន័យមានន័យយ៉ាងណា? ហេតុអ្វីបានជាត្រូវការការបញ្ជូនទិន្នន័យដែលមិនទទួលខុសត្រូវ? តើវាមានន័យគ្រប់គ្រាន់ឬទេ?

សូមស្តាប់ការពន្យល់អំពីភាពខុសគ្នារវាង Protocol ពីរខាងក្រោម:

អ្នកកំពុងអង្គុយជាមួយកុំព្យូទ័រហើយ download ចំពោះ movies។ File មានទំហំ 20GB ហើយក្រោយពី download 10GB បន្ទាប់មកមានបញ្ហាកើតមានឡើងដោយ IP Packets មួយចំនួនមិនបាន download ចូលទៅក្នុងកុំព្យូទ័រ។ នៅពេលអ្នកព្យាយាមចាក់វា អ្នកនឹងឃើញថាមាន Error កើតមានឡើង។ អ្នកមិនអាចមើល movies ហើយអ្នកខឹងដោយផ្តោតអារម្មណ៍ទៅលើ Local DVD ដែលមានគុណភាពមិនល្អ។

ដើម្បីឲ្យការ Download បានល្អ អ្នកត្រូវការប្រើនូវ Reliable Protocol ដែលជា TCP ។ នៅក្នុងករណីខ្លះ IP packets មិនបាន Download ចូលទៅក្នុងកុំព្យូទ័រទេ វាត្រូវតែ Download ឡើងវិញសាជាថ្មី។

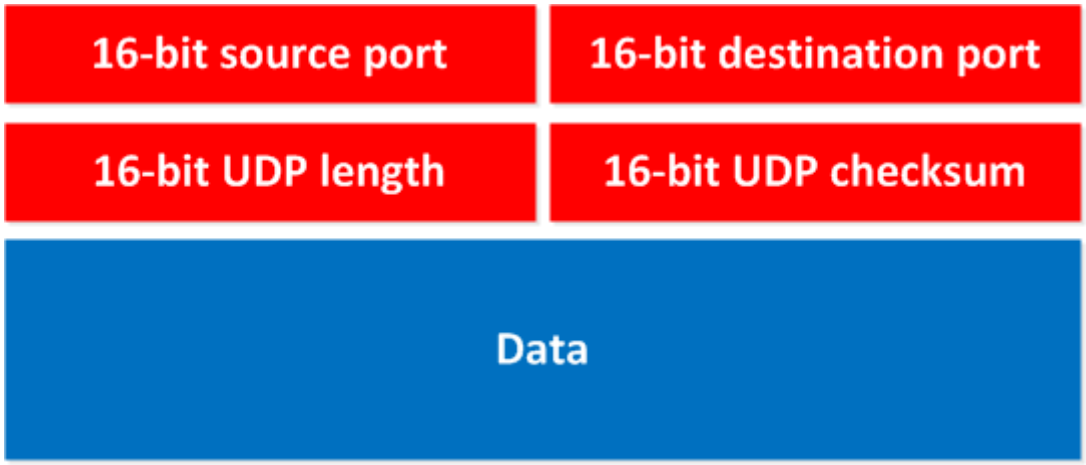
	TCP	UDP
Connection Type:	Connection-oriented	Connectionless
Sequencing:	Yes	No
Usage:	Downloads File Sharing Printing	VoIP Video (streaming)

តើយើងមានអ្វីខ្លះនៅក្នុងតារាងខាងលើ? ជាដំបូងអ្នកបានឃើញពីប្រភេទនៃការភ្ជាប់។ TCP គឺជា connection-Oriented ដែលមានន័យថាវានឹងបង្កើតការភ្ជាប់ហើយបន្ទាប់មកវានឹងចាប់ផ្តើមផ្ទេរទិន្នន័យ។ UDP គឺជា Connectionless ដែលវានិងចាប់ផ្តើមបញ្ជូនដោយមិនចាប់អារម្មណ៍ថាតើវាទៅដល់គោលដៅដែរឬទេ?

ការភ្ជាប់ដែល TCP និងបង្កើតឡើងនោះហៅថា "3 way handshake" ដែលមានដូចខាងក្រោម៖

- បើអ្នកកំពុង Download នូវ File ដ៏ធំមួយ អ្នកត្រូវដឹងថាវាត្រូវបានរៀបតាមលំដាប់ឡើងវិញនៅពេលទទួល។ នេះមានន័យថាវាមានលេខលំដាប់សម្គាល់។ ចំពោះ UDP វិញគ្មានការរៀបចំបែបនេះទេ មានន័យថាវាគ្មានលេខលំដាប់នោះទេ។
- ចុះចំណែកឯ VoIP វិញយ៉ាងណាដែរ? តើយើងមិនត្រូវរៀបវាតាមំដាប់ចំពោះ Packets ទាំងនោះមែនទេនៅខាងឧបករណ៍ទទួល? យើងពិតជាត្រូវការរៀបវាតាមលំដាប់ហើយ បើពុំដូច្នោះទេការសន្ទនាមិនអាចស្តាប់បាននោះទេ។ UDP មិនផ្តល់ឲ្យនូវលេខលំដាប់នោះទេ។ ចំពោះ VoIP វិញ វាមិនប្រើតែ UDP នោះទេ គេប្រើនូវ RTP ដែលជាលេខលំដាប់។

សូមពិនិត្យមើលទៅលើ UDP header ខាងក្រោម:



អ្នកអាចឃើញវាយ៉ាងងាយ វាមានលេខសម្គាល់ពីប្រភពនិងលេខសម្គាល់ពីគោលដៅ(នេះគឺជាការសម្គាល់ពី Application ដែលទិន្នន័យបានប្រើ) វាមាន Checksum និងប្រវែង។

សរុបសេចក្តីមកវិញយើងបានដឹងពី UDP ដូចខាងក្រោម:

វាដំណើរការនៅលើ transport layer នៃ OSI model.

វាជា connectionless protocol ដែលមិនបង្កើតការភ្ជាប់នោះទេ ដោយគ្រាន់តែបញ្ជូនទិន្នន័យប៉ុណ្ណោះ

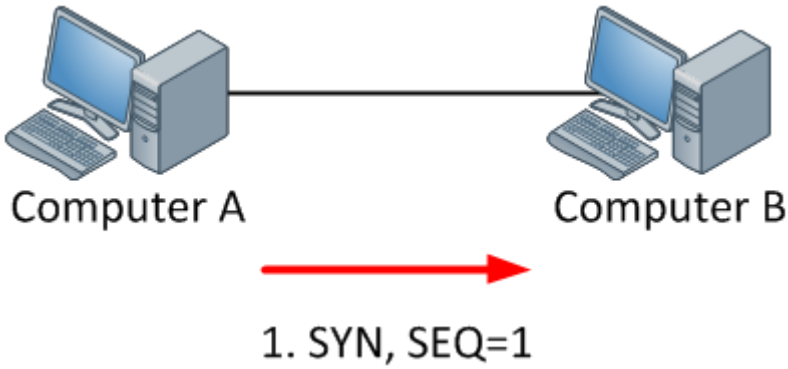
អាចកែ Error បានដោយសារតែមាន Checksum មួយ

វាជា Protocol មិនទទួលខុសត្រូវក្នុងការបញ្ជូនទិន្នន័យ

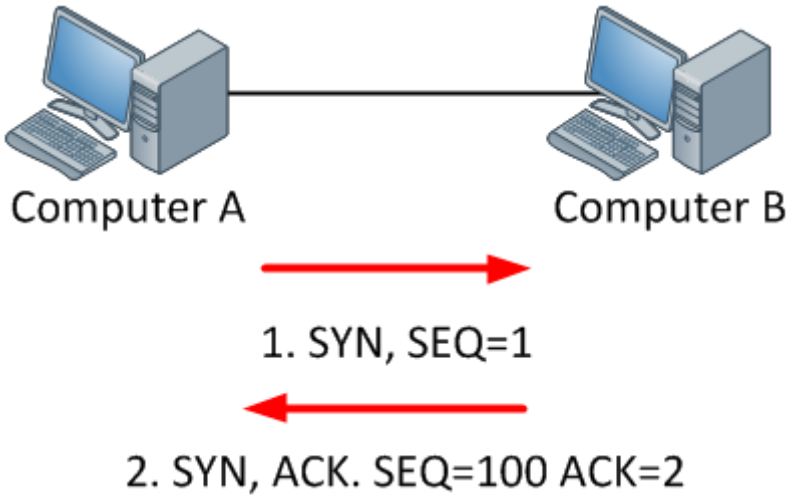
គ្មានលក្ខណៈពិសេសក្នុងសង្គ្រោះទិន្នន័យ

ឥឡូវនេះសូមពិនិត្យមើលពី TCP វិញតើមានផ្តល់អ្វីដល់យើង? ដោយសារតែវាជា Protocol ដែលមានទំនុកចិត្ត វានឹងបង្កើតការភ្ជាប់មុនពេលវាចាប់ផ្តើមធ្វើការបញ្ជូនទិន្នន័យណាមួយ។ ការភ្ជាប់នេះហៅថា "3 way handshake" ។ សូមពិនិត្យមើលទៅលើការពន្យល់ចំពោះកុំព្យូទ័រ២ដែលអ្នកចង់បញ្ជូនទិន្នន័យឲ្យគ្នាទៅវិញទៅមក។

កុំព្យូទ័រ A ចង់បញ្ជូនទិន្នន័យទៅឲ្យកុំព្យូទ័រ B ក្នុងទំនុកចិត្តមួយ ដូច្នេះយើងកំពុងប្រើនូវ TCP ដើម្បីបង្កើតការបញ្ជូននេះប្រព្រឹត្តិទៅបាន។ ជាដំបូងយើងនឹងបង្កើតការភ្ជាប់ដោយប្រើ 3 way handshake ។



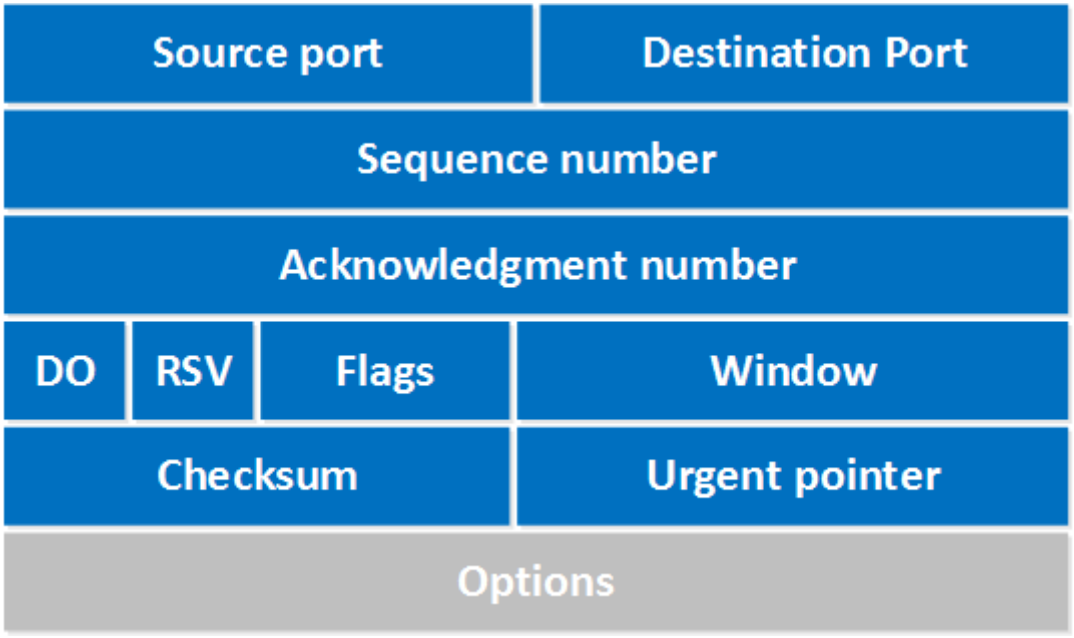
ដំបូងកុំព្យូទ័រ A នឹងបញ្ជូននូវ TCP SYN មួយប្រាប់ទៅឲ្យកុំព្យូទ័រ B បានដឹងថាវាចង់បង្កើតការភ្ជាប់មួយ។ វាក៏មានលេខលំដាប់មួយហើយរក្សាទុកវាដោយប្រើលេខ១។



๑-๔-๑-TCP Header

TCP (Transmission Control Protocol) គឺជា transport protocol ដែលមានទំនុកចិត្តមួយដោយវាបង្កើតការភ្ជាប់មុនពេលធ្វើការបញ្ជូនទិន្នន័យណាមួយហើយអ្វីដែលវាបានបញ្ជូនមកវិញគឺចំលើយឆ្លើយតបវិញដោយឧបករណ៍ទទួល។

នៅក្នុងមេរៀននេះយើងនឹងពិនិត្យទៅលើ TCP header ហើយនិងផ្នែកផ្សេងគ្នា។



ផ្នែកទាំងនេះមានដូចជា:

Source port: មានប្រវែង 16 bits ដែនកំនត់ច្បាស់លាស់ពីលេខ port នៃឧបករណ៍បញ្ជូន។

Destination port: មានប្រវែង 16 bits ដែនកំនត់ច្បាស់លាស់ពីលេខ Port នៃឧបករណ៍ទទួល

Sequence number: គឺជាលេខលំដាប់ដែលមានប្រវែង 32 bits សម្រាប់ចង្អុលបង្ហាញពីទំហំទិន្នន័យដែលកំពុងបញ្ជូនក្នុងកំឡុងពេលនៃ TCP session ។ នៅពេលអ្នកបង្កើតនូវ TCP connection ថ្មីមួយ (3 way handshake) បន្ទាប់មកវាបានចាប់ផ្តើមនូវលេខលំដាប់ចៃដន្យដែលមានប្រវែង 23 bits។ ឧបករណ៍ទទួលនឹងប្រើលេខលំដាប់នេះសម្រាប់បញ្ជូនត្រឡប់មកវិញនូវ ACK មួយ។ Protocol analyzer ដូចជា wireshark នឹងប្រើនូវ relative sequence number នៃ 0 ដោយសារតែវាងាយស្រួលអានជាងលេខដែលមានលំដាប់ខ្ពស់ជាង។

Acknowledgment number: មានប្រវែង 32 bits ត្រូវបានប្រើប្រាស់ដោយឧបករណ៍ទទួលដើម្បីស្នើសុំចំពោះ TCP segment បន្ទាប់។ តម្លៃនេះនឹងក្លាយជាលេខលំដាប់ដែលមានការកើនឡើងម្តង១ជាស្វ័យប្រវត្តិ។

DO: គឺជា data offset ដែលមានប្រវែង 4 bits ហើយគេក៏ហៅផងដែរថាជាប្រវែង header ។ វាបង្ហាញឲ្យដឹងពីប្រវែងនៃ TCP header ។ ដូច្នេះយើងអាចដឹងពីទីកន្លែងដែលទិន្នន័យពិតប្រាកដចាប់ផ្តើម។

RSV: គឺជាផ្នែកបម្រុងទុកដែលមានប្រវែង 3 bits ។ គេមិនប្រើវានោះទេហើយជាទូទៅមានតម្លៃ0 ។

Flags: មានប្រវែង 9 bits។ យើងក៏ហៅវាផងដែរថាជា Control bits។ យើងប្រើវាដើម្បីបង្កើតការភ្ជាប់ បញ្ជូនទិន្នន័យនឹងបញ្ចប់ការភ្ជាប់ទៅវិញ។

URG: គឺជា Pointer បន្ទាន់មួយ។ នៅពេល bit ត្រូវបានកំនត់ទិន្នន័យត្រូវបានចាត់ទុកថាមានអាទិភាពទៅលើទិន្នន័យផ្សេងទៀត។

ACK: ប្រើសម្រាប់ acknowledgment.

PSH: គឺជាអនុគមន៍ push ។ វាប្រាប់ Application មួយថាទិន្នន័យត្រូវតែបញ្ជូនភ្លាមហើយមិនចងរងចាំចំពោះ TCP segment ទាំងមូលនោះទេ ។

RST: សម្រាប់បង្កើតការភ្ជាប់ឡើងវិញ ។ នៅពេលអ្នកទទួលបានវា មានន័យថាវាបានផ្តាច់ការភ្ជាប់ឥឡូវនេះ ។ គេប្រើវាតែក្នុងករណីដែលមាន Error មិនអាចសង្គ្រោះបានហើយវាមិនមែនជាវិធីធម្មតាសម្រាប់បញ្ចប់ TCP connection នោះទេ ។

SYN: យើងប្រើវាសម្រាប់ការចាប់ផ្តើមនៃ 3-way handshake ហើយគេប្រើវាសម្រាប់កំនត់ពីលេខលំដាប់

FIN: គឺជា bit បញ្ចប់ត្រូវបានប្រើដើម្បីបញ្ចប់ TCP connection ។ TCP គឺជា full duplex ដូច្នេះភាគីទាំងពីរនិងប្រើ FIN bit ដើម្បីបញ្ចប់ការភ្ជាប់ ។

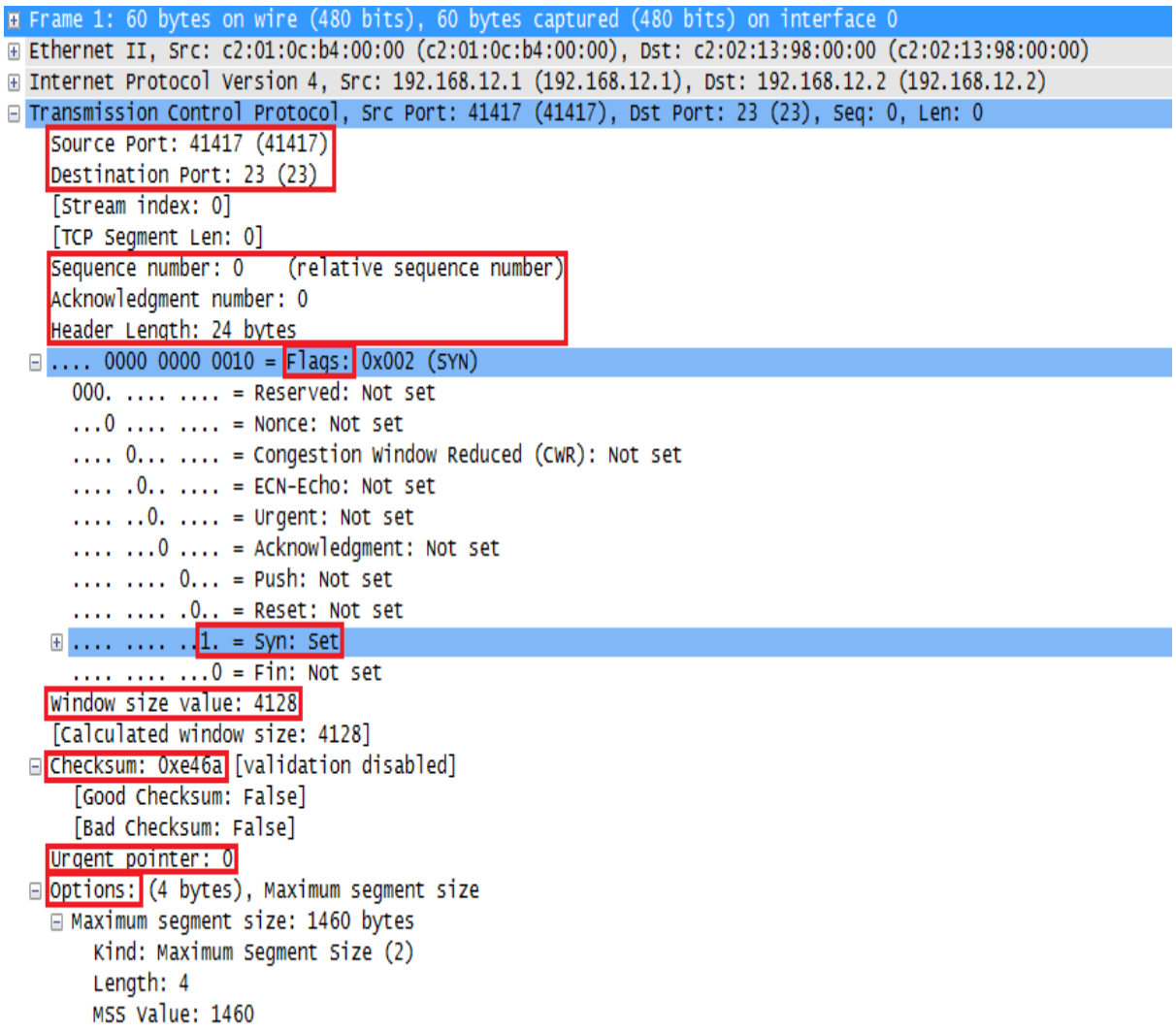
Window: មានប្រវែង 16 bits សម្រាប់បញ្ជាក់ឲ្យច្បាស់លាស់ពីទំហំដែលត្រូវទទួលដោយឧបករណ៍ទទួល ។ គេប្រើវាដើម្បីឲ្យឧបករណ៍ទទួលប្រាប់ឧបករណ៍បញ្ជូនដឹងថាវានឹងទទួលទិន្នន័យច្រើនជាងអ្វីដែលវាកំពុងទទួល ។ វាធ្វើដូច្នេះទៅបានដោយកំនត់ពីចំនួននៃ bytes នៅលើលេខលំដាប់នៅក្នុង acknowledgment field

Checksum: មានប្រវែង 16 bits ត្រូវបានប្រើសម្រាប់ CHECKSUM ដើម្បីត្រួតពិនិត្យថាតើ TCP header មាន Error ឬទេ ។

Urgent pointer: មានប្រវែង 16 bits ហើយត្រូវបានប្រើនៅពេលដែល URG bit ត្រូវបានកំនត់ ។urgent pointer ត្រូវបានប្រើដើម្បីចង្អុលបង្ហាញឲ្យដឹងទិន្នន័យត្រូវបានបញ្ចប់ ។

Options: គឺជា Optional ហើយអាចមានប្រវែងពី 0 ទៅ 230 bits

ដើម្បីពិនិត្យឲ្យដឹងពីវាដែលកំពុងមានសកម្មភាពនោះគឺអ្នកត្រូវប្រើ wireshark ។ ខាងក្រោមនេះគឺជាឧទាហរណ៍នៃ TCP three way handshake ។



ដូចដែលយើងបានឃើញចំពោះលេខ Port របស់ប្រភពបញ្ជូននិងលេខ port របស់គោលដៅដែលត្រូវទទួល។ លេខលំដាប់គឺ0 ប៉ុន្តែ wireshark ប្រាប់យើងថាវាជា relative sequence number មួយ។ តាមការពិតវាជាអ្វីមួយផ្សេង។ អ្នកអាចឃើញពី SYN bit ត្រូវបានកំនត់នៅក្នុង flags, window size, checksum, urgent pointer និង options។

TCP គឺជា protocol ដែលមានលក្ខណៈសាំញ៉ា ប៉ុន្តែវាបានជួយយើងឲ្យអាចយល់ពីអ្វីដែល TCP header មានទម្រង់។

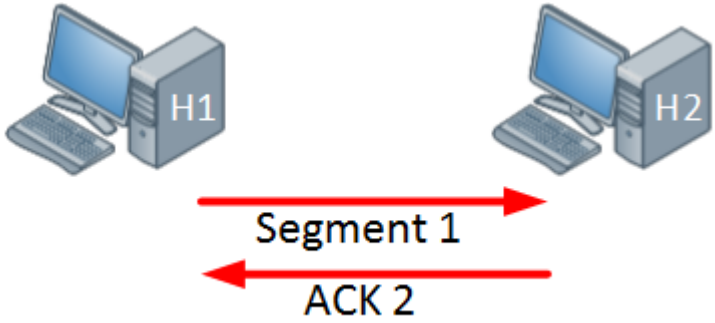
៦-៤-២-TCP Window Size Scaling

TCP (Transmission Control Protocol) គឺជា connection-oriented protocol មានន័យថាយើងបានកត់ត្រាពីទំហំនៃទិន្នន័យដែលបានបញ្ជូន។ ឧបករណ៍បញ្ជូននឹងបញ្ជូនទិន្នន័យខ្លះហើយឧបករណ៍ទទួលក៏បានផ្តល់មកវិញនូវចំលើយឆ្លើយតបវិញ។ នៅពេលយើងមិនបានទទួលមកវិញនូវចំលើយតបវិញនោះទេ នោះឧបករណ៍បញ្ជូននឹងធ្វើការបញ្ជូនឡើងវិញចំពោះទិន្នន័យនោះ។

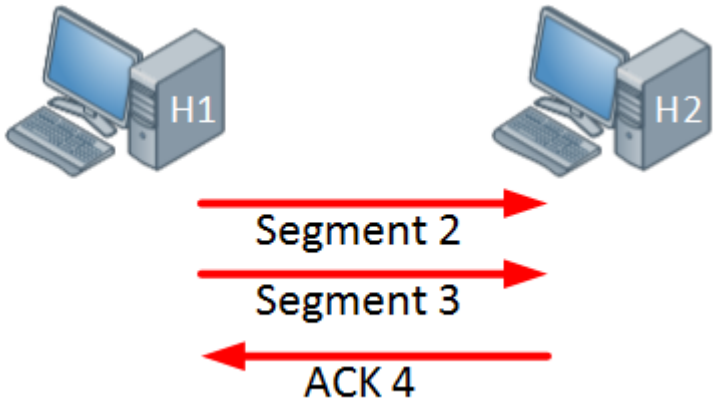
TCP ប្រើ “windowing” មានន័យថាឧបករណ៍បញ្ជូនមួយនិងបញ្ជូននូវទិន្នន័យមួយឬច្រើនហើយឧបករណ៍ទទួលនឹងផ្តល់មកវិញនូវចំលើយឆ្លើយតបមួយឬឆ្លើយតបចំពោះគ្រប់ Segments ទាំងអស់។ នៅពេលដែលយើងចាប់ផ្តើមនូវ

ការភ្ជាប់ TCP មួយ Host នឹងប្រើកន្លែងសម្រាប់រក្សាទុកជាបណ្តោះអាសន្នរបស់ឧបករណ៍ទទួលដែលអាចរក្សាទុកទិន្នន័យជាបណ្តោះអាសន្នមុនពេល Application ដំណើរការវា។ នៅពេលឧបករណ៍ទទួលបញ្ជូនមកវិញនូវចំលើយតបវិញ វានិងប្រាប់ឲ្យឧបករណ៍បញ្ជូនដឹងពីទំហំទិន្នន័យដែលវាអាចបញ្ជូនមុនពេលឧបករណ៍ទទួលនឹងផ្តល់មកវិញនូវចំលើតតបវិញ។ យើងហៅវាថាជា Windows size ។ ជាមូលដ្ឋានគ្រឹះ Windows size បង្ហាញពីទំហំនៃកន្លែងរក្សាទុកជាបណ្តោះអាសន្នរបស់ឧបករណ៍ទទួល។

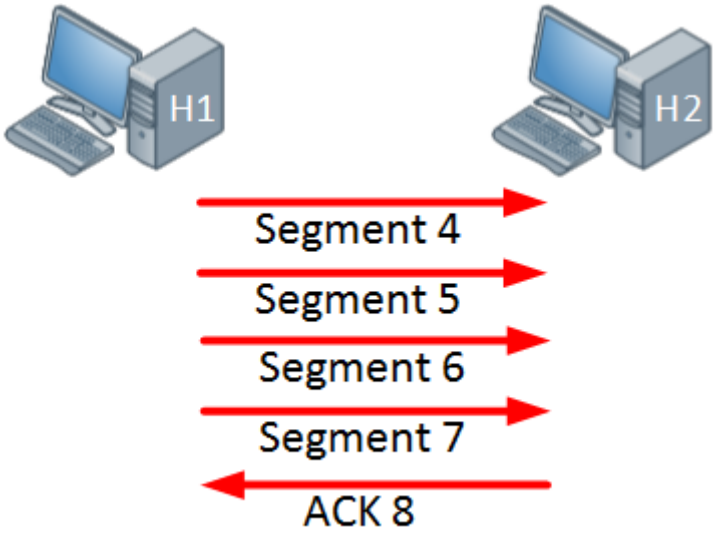
នៅពេលដែលមានការភ្ជាប់ចំពោះ TCP វានឹងមាន Windows size មួយដែលមានទំហំតូចហើយគ្រប់ពេលដែលទទួលបាននូវចំលើយតបវិញទទួលបានជោគជ័យមួយ Windows size មានទំហំកើនឡើង។ ខាងក្រោមនេះជាឧទាហរណ៍



Hosts ពីរខាងលើគឺ Host នៅខាងឆ្វេងនឹងបញ្ជូននូវ Segment មួយរហើយ Host ខាងស្តាំនិងបញ្ជូនមកវិញនូវចំលើយតបវិញ។ ដោយសារតែចំលើយតបវិញទទួលបានជោគជ័យ នោះ Windows Size និងមានការកើនឡើងទំហំ។



Host នៅខាងឆ្វេងកំពុងបញ្ជូននូវ Segment ពីរហើយ Host ខាងស្តាំនិងបញ្ជូនត្រឡប់មកវិញនូវចំលើយតបវិញមួយ។ អ្វីដំណើរការបានល្អ ដូច្នេះ Windows Size និងមានការកើនឡើងបន្ថែមទៀត។



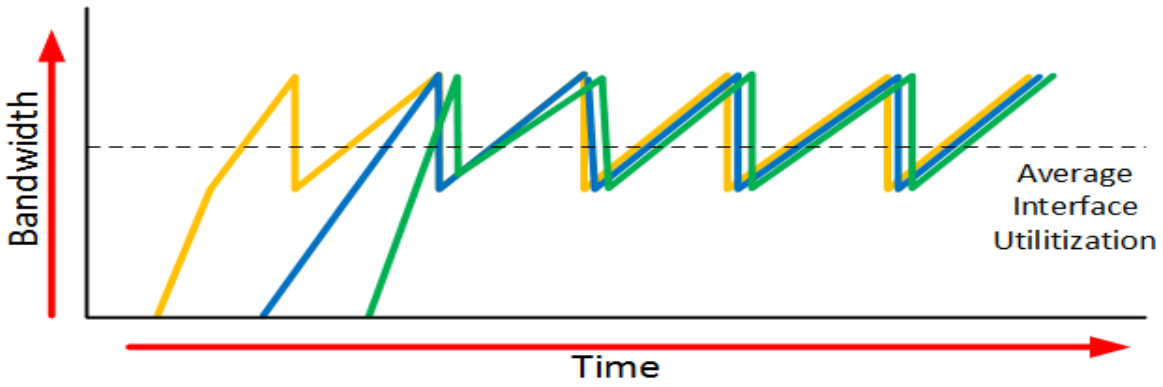
ឥឡូវនេះ Host កំពុងបញ្ជូននូវ 4 segments ហើយ Host នៅខាងស្តាំកំពុងបញ្ជូនមកវិញនូវចំលើយតបវិញតែមួយគត់ ។

នៅក្នុងឧទាហរណ៍ខាងលើ Windows Size មានការកើនឡើងពីទំហំនៅពេលដែលឧបករណ៍ទទួលបញ្ជូនត្រឡប់វិញនូវចំលើយតបវិញចំពោះ 4 segments ឬនៅពេលដែល Windows Size បានកើនឡើងដល់កម្រិតអតិបរមា ។ នៅពេលដែលឧបករណ៍ទទួលមិនបានបញ្ជូនត្រឡប់វិញនូវចំលើយតបវិញទេក្នុងរយៈពេលមួយ (ហៅថារយៈពេលវិញជុំ) នោះ Windows Size និងមានទំហំថយចុះ ។

នៅពេល Interface មានជួបនូវបញ្ហាកកស្ទះ នោះវាបោះបង់ចោលនូវ TCP Packets ខ្លះ ។ ដើម្បីដោះស្រាយបញ្ហានេះ TCP មានអាល់ហ្គូរីតមួយចំនួនដែលគ្រប់គ្រងទៅលើការកកស្ទះនេះ ។ អាល់ហ្គូរីតមួយក្នុងចំណោមអាល់ហ្គូរីតទាំងនោះគឺ slow start ។

ការកកស្ទះ (Congestion) កើតឡើងនៅពេល Interface បានបញ្ជូនទិន្នន័យច្រើនជាងវាអាចបម្រើបាន ។ វាមានការរៀបចំបន្តកន្ទុយគ្នារហូតដល់ពេញហើយ Packets និងត្រូវបោះបង់ចោល ។

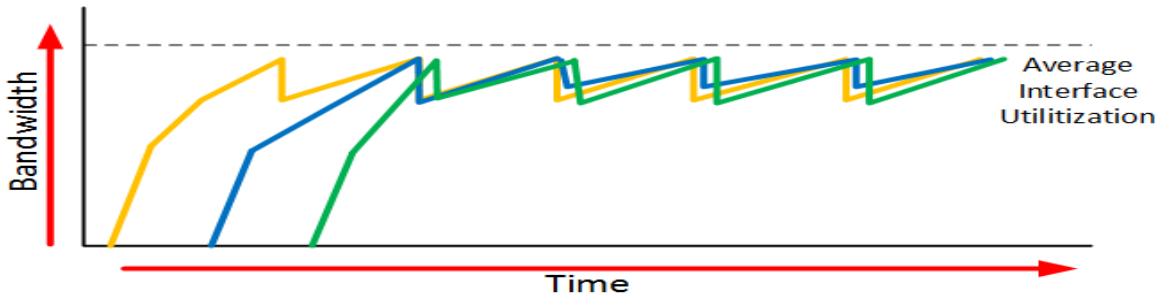
ជាមួយ TCP slow start window size និងមានការកើនឡើងទ្វេដង ប៉ុន្តែនៅពេលដែល Packets ត្រូវបានបោះចោលភ្លាម នោះ Windows Size និងមានទំហំថយចុះមួយ Segment ។ បន្ទាប់មកវាមានការកើនឡើងវិញ2ដងភ្លាមរហូតទាល់តែ window size មានកកស្ទះពាក់កណ្តាលកើតមានឡើង ។ នៅពេលដែល Interface ទទួលបានជួបនូវការកកស្ទះ នោះគ្រប់ការភ្ជាប់នៃ TCP និងមានបទពិសោធន៍ជាមួយ TCP slow start ។ Packets និងត្រូវបានបោះបង់បោលហើយបន្ទាប់មកគ្រប់ការភ្ជាប់ TCP ទាំងអស់មានទំហំ Windows Size តូច ។ នេះគេហៅវាថាជា TCP global synchronization ។ ខាងក្រោមនេះជាឧទាហរណ៍



ពណ៌លឿង ខៀវនិងបៃតងគឺជាការភ្ជាប់នៃ TCP បីប្រភេទផ្សេងគ្នា។ ការភ្ជាប់នៃ TCP ទាំងនេះចាប់ផ្តើមនៅពេលវេលាផ្សេងគ្នាហើយក្រោយមក Interface ក៏ជួបនូវការកកស្ទះហើយ Packets នៃការភ្ជាប់របស់ TCP ទាំងអស់ក៏វាបានបោះបង់ចោលទាំងអស់។ តើមានអ្វីកើតមានឡើងនៅពេលដែល Windows Size នៃការភ្ជាប់ TCP ទាំងនេះនិងបោះបង់បោលហើយក្លាមនោះការកកស្ទះនៃ Interface ក៏កើតមានឡើង នោះទំហំនៃ Windows Size ក៏កើនឡើងម្តងទៀត។

បន្ទាប់មក Interface ក៏ទទួលបាននូវការកកស្ទះម្តងទៀត Windows Size ក៏បោះបង់ចោលហើយវាក៏មាន Windows Size កើនឡើងវិញ។ លទ្ធផលគឺថាយើងមិនប្រើទំហំនៃ bandwidth ទាំងអស់នោះទេ។ បើអ្នកពិនិត្យមើលទៅលើបន្ទាត់គូសដាច់ៗអ្នកនិងឃើញថាការប្រើប្រាស់ជាមធ្យមនៃ Interface មិនខ្ពស់នោះទេ។

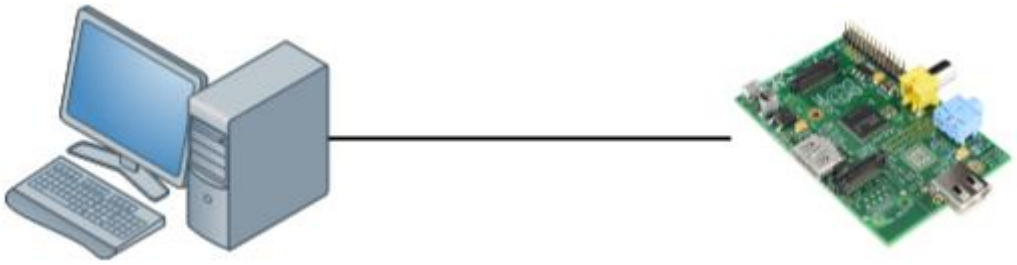
ដើម្បីការពារ global synchronization យើងប្រើ RED (Random Early Detection)។ នេះគេហៅថាលក្ខណៈពិសេសដែលបោះបង់ចោលដោយចៃដន្យចំពោះ Packets ពី TCP ដោយពឹងផ្អែកទៅលើចំនួននៃ Packets នៅក្នុងការបន្តកន្ទុយគ្នាហើយ TOS (Type of Service) ធ្វើការសម្គាល់ទៅលើ Packets។ នៅពេលដែល Packets ត្រូវបានបោះចោលមុនពេលដែលការបន្តកន្ទុយគ្នាពេញ នោះយើងអាចជៀសវាងចំពោះ global synchronization។ លទ្ធផលមានដូចខាងក្រោម:



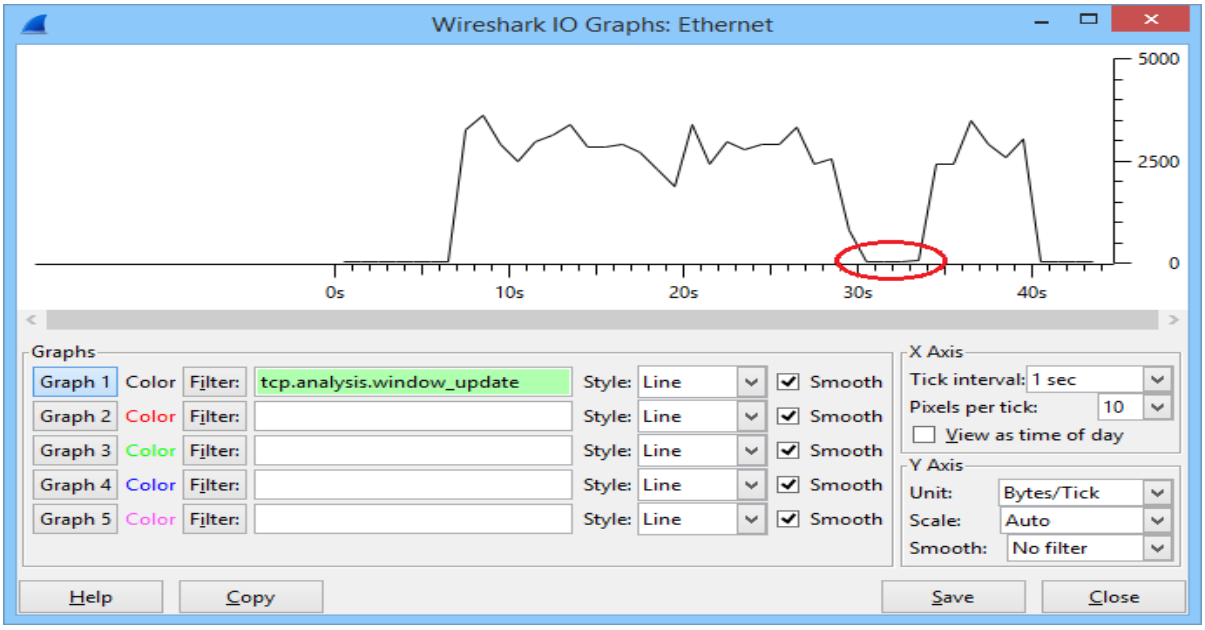
នៅពេលយើងប្រើ RED នោះការប្រើប្រាស់នៃ Interface ជាមធ្យមនឹងមានការល្អប្រសើរ។

៦-៥-Wireshark Captures

សូមពិនិត្យមើលពី TCP window size រវាងឧបករណ៍ពីរ:



ឧបករណ៍នៅខាងឆ្វេងគឺជាកុំព្យូទ័រមួយមាន gigabit interface។ ចំណែកឯខាងស្តាំវិញគឺជា raspberry pi ដែលមាន FastEthernet interface មួយ។ raspberry pi គឺជាឧបករណ៍តូចមួយ ប៉ុន្តែវាមាន cpu/memory/ ethernet interface ។ ដើម្បីទទួលបានលទ្ធផល យើងនិងថតចំលងនូវ File ធំមួយតាមរយៈនៃ SSH ពីកុំព្យូទ័រទៅឲ្យ raspberry pi ។ សូមពិនិត្យមើលរូបភាពខាងក្រោម:

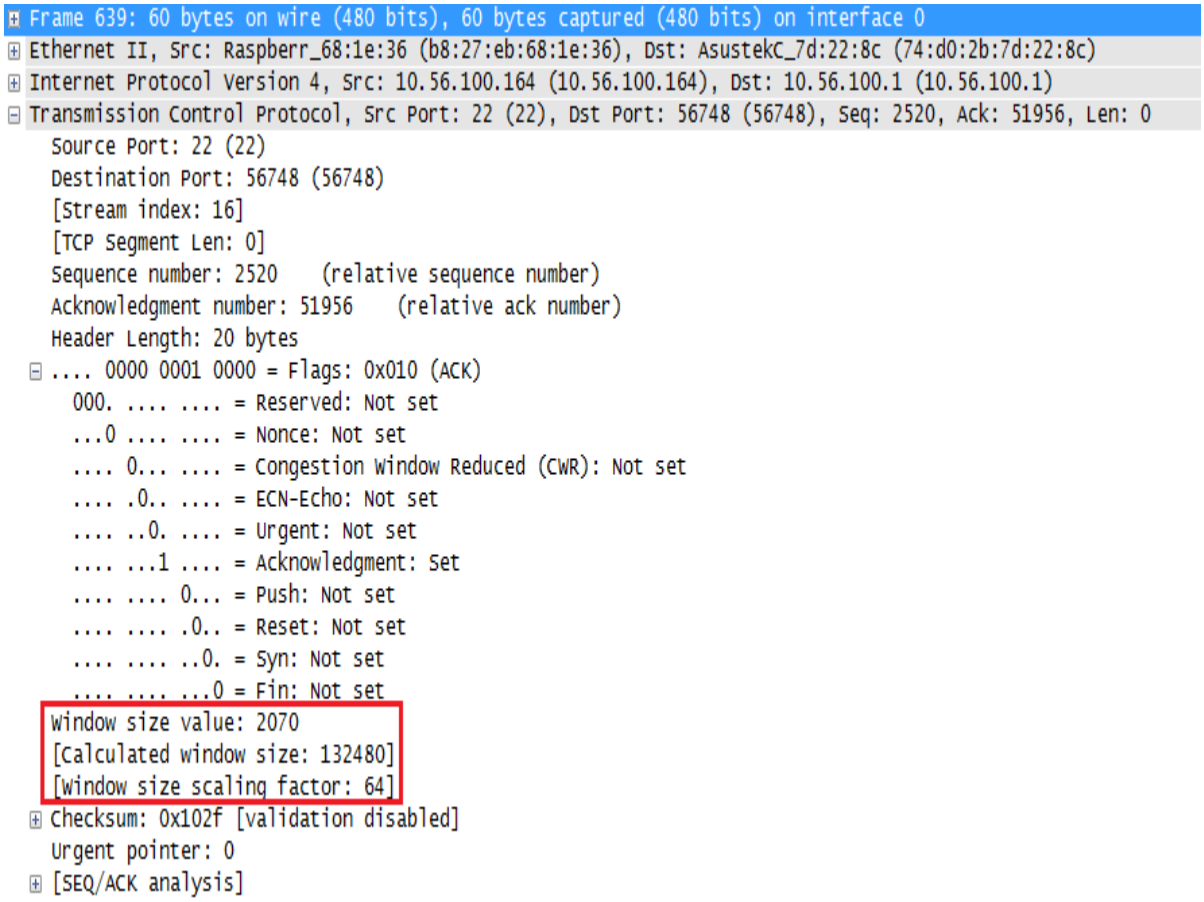


នៅក្នុងក្រាហ្វិចខាងលើ អ្នកបានឃើញថា Windows Size ត្រូវបានប្រើក្នុងការភ្ជាប់នេះ។ ការផ្ទេរ File បានចាប់ផ្តើមក្រោយពី៦វិនាទីហើយអ្នកក៏ឃើញថា Windows Size កើនឡើងយ៉ាងឆាប់រហ័ស។ វាមានការកើនឡើងនឹងចុះវិញបន្តិច ប៉ុន្តែ 30 វិនាទីក្រោយវាបានធ្លាក់ចុះវិញ។ ក្រោយពីពីរបីវិនាទីមក វាមានការកើនឡើងវិញ ហើយការផ្ទេរ File ក៏បានបញ្ចប់ដែរ។ សូមពិនិត្យមើលរូបភាពខាងក្រោមនេះដែលទាក់ទងជាមួយ 3 way handshake ។

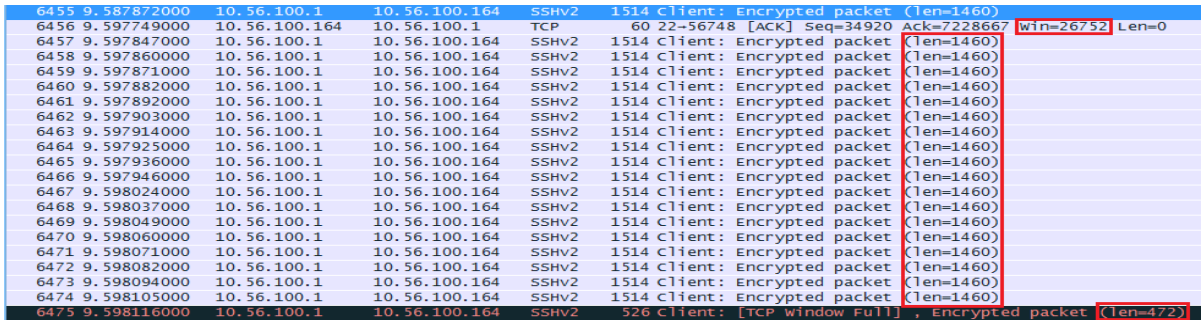
No.	Time	Source	Destination	Protocol	Length	Info
475	6.531678000	10.56.100.1	10.56.100.164	TCP	66	56748→22 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=128 SACK_PERM=1
476	6.532095000	10.56.100.164	10.56.100.1	TCP	66	22→56748 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=64
477	6.532144000	10.56.100.1	10.56.100.164	TCP	54	56748→22 [ACK] Seq=1 Ack=1 Win=4194304 Len=0

កុំព្យូទ័រមាន IP address 10.56.100.1 ហើយ raspberry pi មាន IP address 10.56.100.164។ ដូចអ្នកបានឃើញខាងលើ នៅក្នុង SYN,ACK message ដែល raspberry piប្រើ window sizeគឺ29200 ។

ចំណែកឯកុំព្យូទ័រប្រើ window size គឺ 4194304 ដែលមិនត្រូវគ្នា។ ក្រោយពី packets មួយចំនួន Windows Size របស់ raspberry pi បង្ហាញដូចខាងក្រោម:

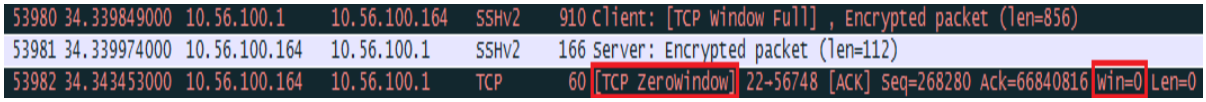


ដូចអ្នកបានឃើញខាងលើស្រាប់ Windows Size បានកើនឡើងរហូតដល់ 132480។ the window size គឺមានទំហំ 16 bits ដែលអាចមានទំហំធំបំផុតគឺ 65535។ សព្វថ្ងៃនេះយើងប្រើ Scaling factor ដែលអាចពង្រីកទំហំ Windows Size បានធំ។ ប្រហែលជា 10 វិនាទីក្រោយមក Windows Size មានការថយចុះវិញ។



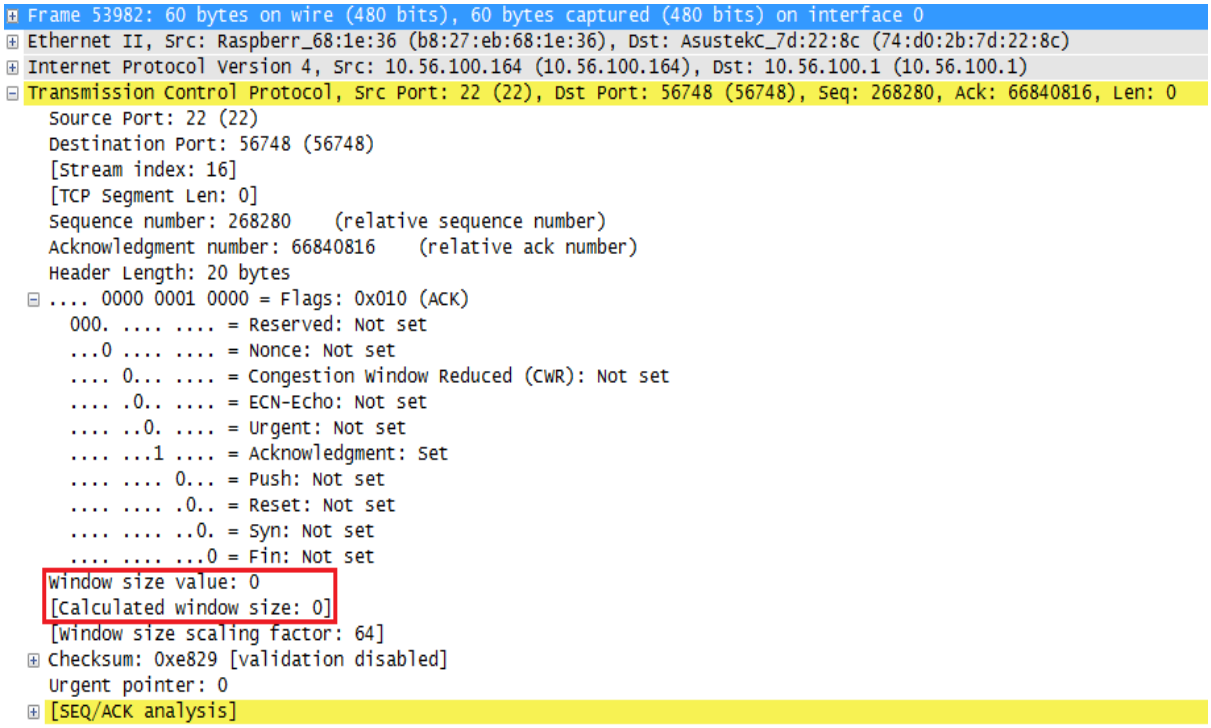
raspberry pi បានរក្សាទុកទំហំកើនឡើង២ដងហើយកន្លែងរក្សាទុកបណ្តោះអាសន្នរបស់វាក៏ពេញ។ វាបានប្រាប់កុំព្យូទ័រឲ្យប្រើ Windows Size ដែលមានទំហំ 26752 ចាប់ពីពេលនេះទៅ។ កុំព្យូទ័របញ្ជូន 18 segments ដែលមានទំហំ 1460 bytes ហើយ 1 segment គឺ 472 bytes (សរុបគឺ26752 bytes)។

Packet ចុងក្រោយបង្ហាញឲ្យដឹងពី “TCP Window Full” message។ នេះគឺជាអ្វីដែល wireshark បង្ហាញជារបាយការណ៍ឲ្យយើងដឹងថាកុំព្យូទ័ររបស់យើងបានបំពេញចំពោះកន្លែងសម្រាប់រក្សាទុកជាបណ្តោះអាសន្ននៃ raspberry pi។ នៅពេលដែល raspberry pi បានចាប់យកនូវ bit មួយហើយក្នុងរយៈពេល៣០វិនាទី មានអ្វីមិនល្អបានកើតមានឡើង។ សូមពិនិត្យមើលអ្វីដែល wireshark បានចាប់យក:



ខាងលើបានបង្ហាញឲ្យដឹងថា raspberry pi បានបញ្ជូននូវ ACK មួយមកកាន់កុំព្យូទ័រដែលមាន Windows Size ស្មើ0 ។ នេះមានន័យថា Windows Size នឹងនៅតែមានតម្លៃ0ក្នុងមួយរយៈពេល។ raspberry pi មិនអាចទទួលបាននូវទិន្នន័យបន្ថែមទៀតទេនៅពេលនេះហើយការបញ្ជូន TCP និងផ្អាកមួយរយៈពេលនៅខណៈដែលកន្លែងរក្សាទុកជាបណ្តោះអាសន្នកំពុងដំណើរការ។

នេះគឺជា Packet ពិតប្រាកដមួយ:



ដូចអ្នកបានឃើញខាងលើស្រាប់ Windows Size ឥឡូវនេះមានតម្លៃ0 ។ នៅពេលដែលកន្លែងរក្សាទុកជាបណ្តោះអាសន្នរបស់អ្នកទទួលបានកំពុងដំណើរការនោះ raspberry pi និងបញ្ជូននូវ ACK មួយជាមួយ Windows Size ថ្មីមួយ:

```
Frame 3659: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
Ethernet II, Src: Raspberr_68:1e:36 (b8:27:eb:68:1e:36), Dst: AsustekC_7d:22:8c (74:d0:2b:7d:22:8c)
Internet Protocol Version 4, Src: 10.56.100.164 (10.56.100.164), Dst: 10.56.100.1 (10.56.100.1)
Transmission Control Protocol, Src Port: 22 (22), Dst Port: 56748 (56748), Seq: 23672, Ack: 4472703, Len: 0
    Source Port: 22 (22)
    Destination Port: 56748 (56748)
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 23672    (relative sequence number)
    Acknowledgment number: 4472703    (relative ack number)
    Header Length: 20 bytes
  .... 0000 0001 0000 = Flags: 0x010 (ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
      .... .... ..0. = Syn: Not set
      .... .... ...0 = Fin: Not set
    Window size value: 400
    [Calculated window size: 25600]
    [Window size scaling factor: 64]
  Checksum: 0x4f46 [validation disabled]
    Urgent pointer: 0
  [SEQ/ACK analysis]
```

window size ឥឡូវនេះមានទំហំស្មើនឹង 25600 bytesប៉ុណ្ណោះប៉ុន្តែវាអាចកើនឡើងម្តងទៀត។

៦-៦-ICMP (Internet Control Message Protocol)

ICMP (Internet Control Message Protocol) គឺជា ណេតវើកProtocol មួយត្រូវបានប្រើសម្រាប់ស្វែងរកកំហុសនិងគ្រប់គ្រងទៅលើ Network។ ឧទាហរណ៍ដ៏ល្អទាក់ទងជាមួយវាគឺ “ping” ដែលប្រើ ICMP request និង ICMP reply messages។ នៅពេល Host មួយមិនអាចភ្ជាប់មកកាន់បាន ICMP អាចបញ្ជូននូវ error message មួយទៅឲ្យប្រភពដើមវិញ។ ឧទាហរណ៍មួយទៀតនៃ application ដែលប្រើ ICMP គឺ traceroute។

ICMP messages ត្រូវបាន encapsulate នៅក្នុង IP Packets ដែលមនុស្សភាគច្រើននិយាយថាវាជា layer 4 protocol ដូចជា UDP ឬ TCP។ ទោះបីជាយ៉ាងណាក៏ដោយ ដោយសារតែ ICMP គឺជាផ្នែកមួយនៃ IP protocol វាត្រូវបានគេចាត់ទុកថាជា layer 3 protocol។

Header ដែល ICMP ប្រើគឺមានភាពងាយស្រួលណាស់ នេះគឺជាឧទាហរណ៍:

Byte ទី១បញ្ជាក់ពីប្រភេទនៃ ICMP message។ ឧទាហរណ៍ type 8 ត្រូវបានប្រើសម្រាប់ ICMP request ហើយ type 0 ត្រូវបានប្រើសម្រាប់ ICMP reply។ យើងប្រើ type 3 សម្រាប់គោលដៅដែលមិនបានទៅដល់។

ចំពោះ byte ទី២វិញហៅថា Code បញ្ជាក់ច្បាស់លាស់ពីប្រភេទនៃ ICMP message។ ឧទាហរណ៍ message សម្រាប់គោលដៅដែលមិនដល់គោលដៅមាន 16 codes ខុសគ្នា។ នៅពេលអ្នកឃើញ code 0 មានន័យថាគោលដៅ ណោតរើកមិនអាចទៅដល់នៅទេនៅខណៈពេលដែល Code 1 មានន័យថាគោលដៅ Host មិនអាចទាក់ទងបាន។

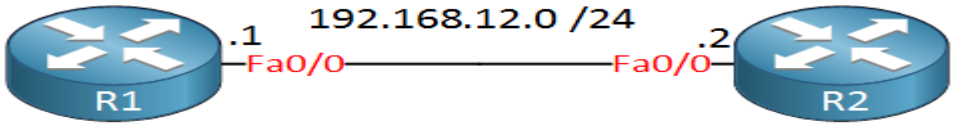
Byte ទី៣គឺមានប្រវែង 2 bytes ត្រូវបានប្រើសម្រាប់ CHECKSUM ដើម្បីត្រួតពិនិត្យឲ្យដឹងថា ICMP header មានខូចឬអត់។ អ្វីដែលនៅសល់នៃHeader អាស្រ័យទៅលើ ICMP messages ដែលយើងកំពុងប្រើវា។

ដើម្បីបង្ហាញឲ្យដឹងកាន់តែច្បាស់ពី ICMP កំពុងមានសកម្មភាព សូមពិនិត្យមើល ICMP messages នៅក្នុង Wireshark។

៦-៧-ការចាប់យករបស់ Wireshark

ICMP Echo request and reply

សូមពិនិត្យមើលជាមួយឧទាហរណ៍ដ៏សាមញ្ញមួយគឺ ping ។



យើងចាប់ផ្តើម ping ចេញពី R1

R1#ping 192.168.12.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:

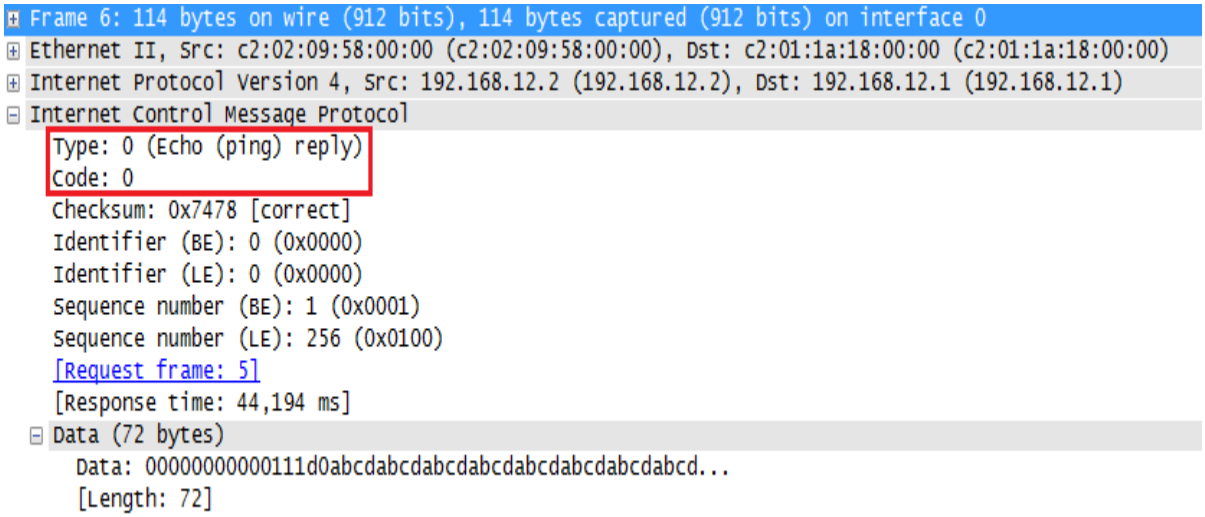
!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/21/64 ms

នេះគឺជាអ្វីដែលយើងបានឃើញ:

```
Frame 5: 114 bytes on wire (912 bits), 114 bytes captured (912 bits) on interface 0
Ethernet II, Src: c2:01:1a:18:00:00 (c2:01:1a:18:00:00), Dst: c2:02:09:58:00:00 (c2:02:09:58:00:00)
Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x6c78 [correct]
    Identifier (BE): 0 (0x0000)
    Identifier (LE): 0 (0x0000)
    Sequence number (BE): 1 (0x0001)
    Sequence number (LE): 256 (0x0100)
    [Response frame: 6]
  Data (72 bytes)
      Data: 00000000000111d0abcdabcdabcdabcdabcdabcdabcdabcd...
      [Length: 72]
```

message ខាងលើគឺជា ICMP request អ្នកអាចឃើញវាជា type 8 ហើយ code 0 ។ នៅពេល R2 ទទួលវានឹងឆ្លើយតបវិញ:



ICMP echo reply គឺជា type 0 និង code 0 message.

ភាពមិនអាចទៅដល់គោលដៅ(Destination Unreachable)

ឧទាហរណ៍ដ៏ល្អមួយទៀតចំពោះសារដែលបញ្ជាក់ថាគោលដៅមិនអាចទៅដល់គឺសូមមើលពីឧទាហរណ៍ខាងក្រោមនេះ។ យើងអាចសាកល្បងជាមួយវាដោយបន្ថែម access-list មួយនៅលើ R2 ដែលបដិសេធចំពោះ ICMP messages:

R2(config)#ip access-list extended NO_ICMP

R2(config-ext-nacl)#deny icmp any host 192.168.12.2

R2(config-ext-nacl)#permit ip any any

R2(config)#interface FastEthernet 0/0

R2(config-if)#ip access-group NO_ICMP in

ឥឡូវនេះសាកល្បង ping ពី R1 វិញ:

R1#ping 192.168.12.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:

UUUUU

Success rate is 0 percent (0/5)

ping បរាជ័យហើយអ្នកអាចឃើញ U (unreachable) messages នៅលើ R1។ នេះជាលទ្ធផល ICMP message ដែល R2 បញ្ជូន:

```
Frame 2: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0
Ethernet II, Src: c2:02:09:58:00:00 (c2:02:09:58:00:00), Dst: c2:01:1a:18:00:00 (c2:01:1a:18:00:00)
Internet Protocol Version 4, Src: 192.168.12.2 (192.168.12.2), Dst: 192.168.12.1 (192.168.12.1)
Internet Control Message Protocol
    Type: 3 (Destination unreachable)
    Code: 13 (Communication administratively filtered)
    Checksum: 0xc54b [correct]
Internet Protocol Version 4, Src: 192.168.12.1 (192.168.12.1), Dst: 192.168.12.2 (192.168.12.2)
    Version: 4
    Header Length: 20 bytes
    Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 100
    Identification: 0x000a (10)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 254
    Protocol: ICMP (1)
    Header checksum: 0x233b [validation disabled]
    Source: 192.168.12.1 (192.168.12.1)
    Destination: 192.168.12.2 (192.168.12.2)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x2fa5
    Identifier (BE): 2 (0x0002)
    Identifier (LE): 512 (0x0200)
    Sequence number (BE): 0 (0x0000)
    Sequence number (LE): 0 (0x0000)
```

ICMP destination unreachable message គឺជា type 3 ហើយប្រើ code 13 ពីព្រោះថា packet នេះគឺជា “administratively filtered” (acces-list) ។

Traceroute

Traceroute ក៏ប្រើ ICMP messages ផងដែរ។ ដើម្បីបង្ហាញឲ្យបានដឹងយើងប្រើ Routers ចំនួន៣ ។

សូមពិនិត្យមើលពីអ្វីដែល traceroute ពី R1 ទៅ R3 :

R1#traceroute 192.168.23.3 probe 1

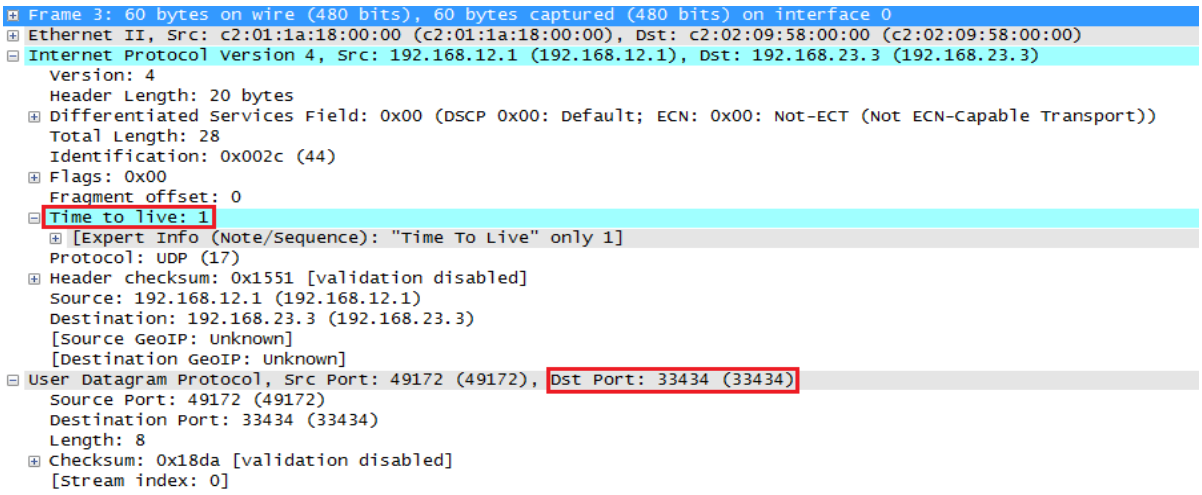
Type escape sequence to abort.

Tracing the route to 192.168.23.3

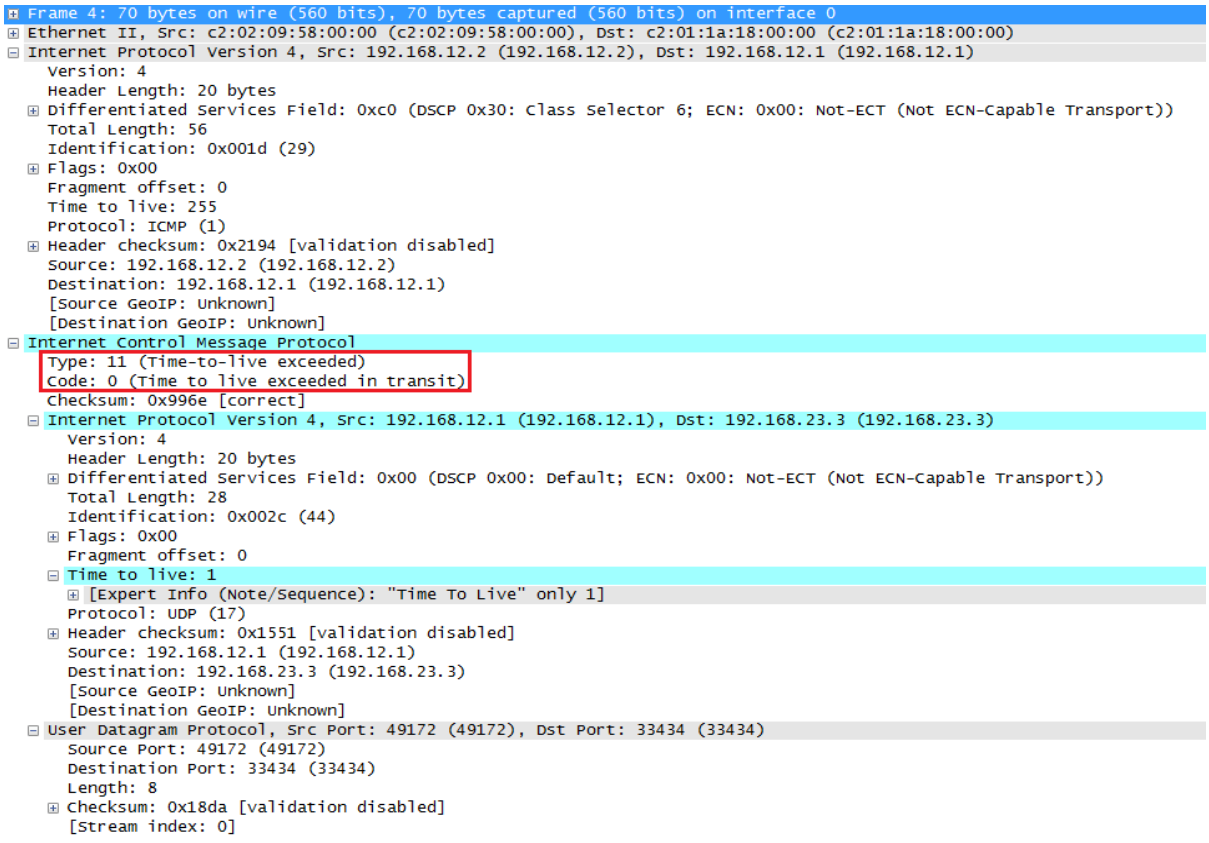
1 192.168.12.2 52 msec

2 192.168.23.3 60 msec

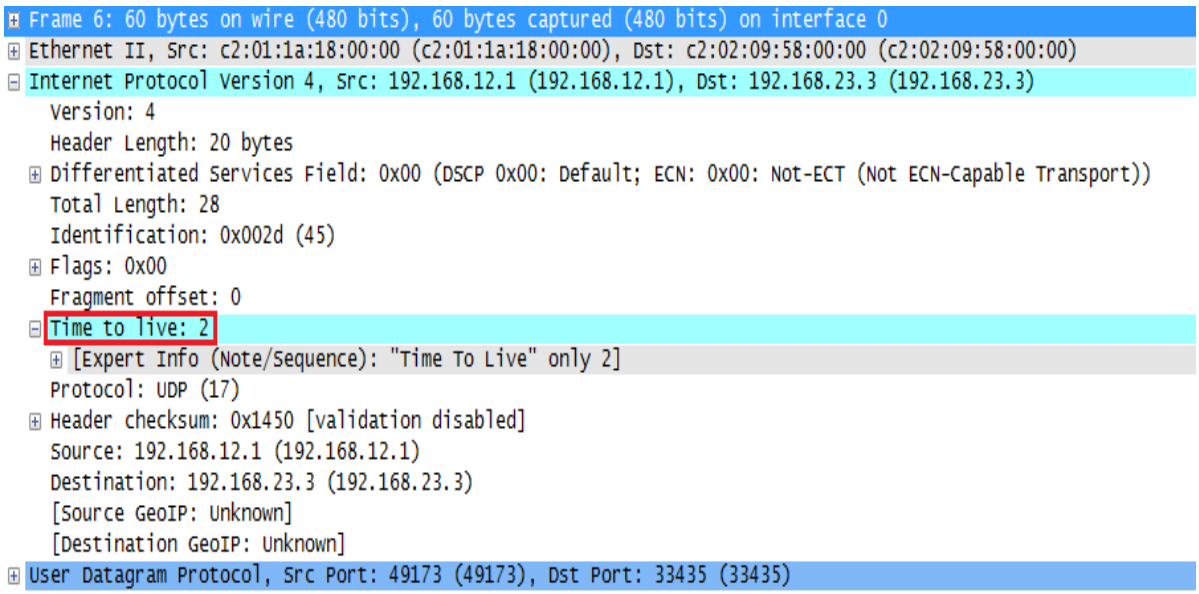
តាមលំនាំដើម ចំពោះ Cisco IOS និងបញ្ជូននូវកស្តុតាងជាច្រើន ។ ចំពោះការបង្ហាញនេះត្រូវការតែកស្តុតាងតែមួយគត់ ។ នេះគឺជា Packet ទី១ដែល R1 បញ្ជូន:



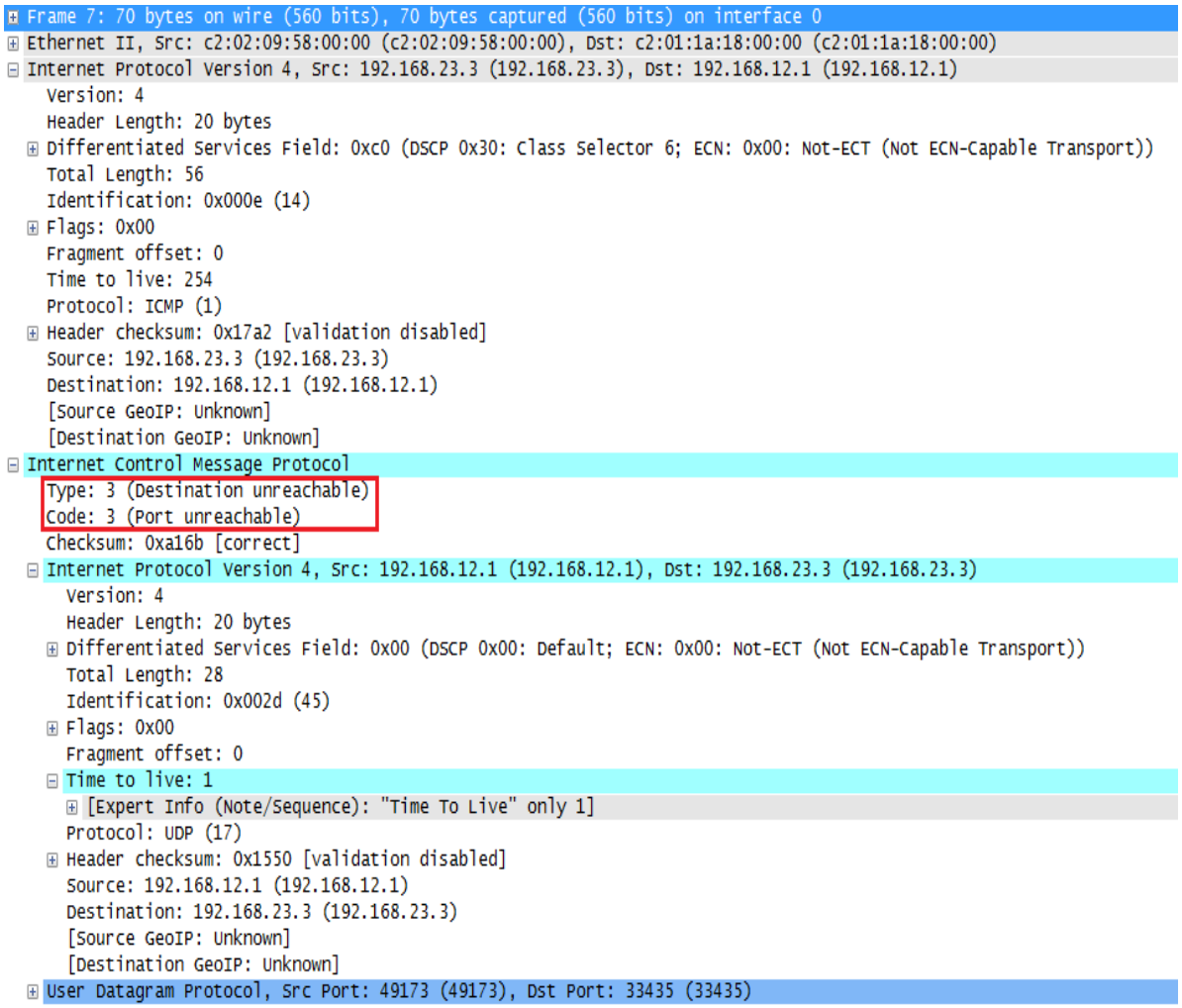
Cisco IOS ប្រើ UDP packets ដែល TTL មានតម្លៃ 1 ហើយ Port ជាគោលដៅគឺ 33434។ TTL នឹងគោលដៅ Port នឹងកើនឡើងនៅគ្រប់ Hop ដែលវាឆ្លងកាត់។ នៅពេលដែល R2 ទទួលបាន packets នេះវានឹងឆ្លើយតបមកវិញដូចខាងក្រោម:



R2 នឹងបញ្ជូននូវ ICMP មួយជា type 11 (time to live exceeded) message ទៅឲ្យ R1។ នៅពេលដែល R1 ទទួលវាភ្លាម វានិងបញ្ជូននូវកេស្តុតាងទី២:



ដូចអ្នកបានឃើញខាងលើស្រាប់ TTL គឺមានតម្លៃស្មើ 2 ហើយ Port ជាគោលដៅបានកើនឡើងដល់ 33435។ នៅពេលដែល R3 ទទួលបាន packet នេះវានិងឆ្លើយតបវិញដូចខាងក្រោម:



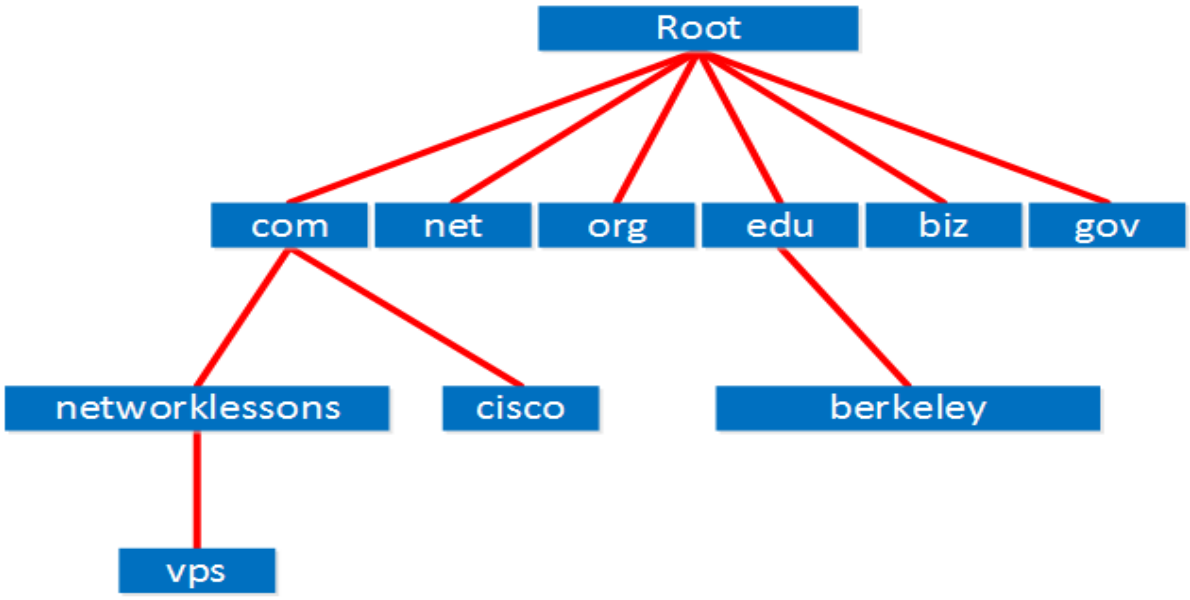
R3 នឹងឆ្លើយតបវិញជាមួយ type 3 destination unreachable message។ សូមពិនិត្យមើល code វាបង្ហាញឲ្យដឹងថា port មានអាចភ្ជាប់បាន។ ពីព្រោះថាគ្មានអ្វីកំពុងស្តាប់នៅលើ UDP port 33435 យ៉ាងហោចណាស់ R1 បានដឹងថា IP address 192.168.23.3 មិនអាចភ្ជាប់បាន។

១-៨-ការណែនាំឲ្យស្គាល់ DNS

DNS (Domain Name System) គឺជា ណេតវើកprotocol មួយដែលយើងប្រើសម្រាប់រកឲ្យឃើញពី IP address របស់ Hostname។ កុំព្យូទ័រប្រើ IP address ប៉ុន្តែសម្រាប់មនុស្សវិញប្រើឈ្មោះ (hostname) ជំនួសឲ្យ IP address ។

DNS គឺជា distributed and hierarchical ដែលមាន DNS servers រាប់ពាន់ ប៉ុន្តែពួកវាគ្មាន Database ទាំងស្រុងនោះទេជាមួយ hostnames/domain names និង IP addresses។ DNS server មួយអាចមានព័ត៌មានសម្រាប់ domains ខ្លះ ប៉ុន្តែអាច query មកកាន់ DNS server ផ្សេងទៀតបើមិនបានឆ្លើយតបវិញនូវចំលើយ។

មាន root name servers ទាំងអស់ចំនួន១៣ ដែលមានព័ត៌មានសម្រាប់ generic top level domains ដូចជា com, net, org, biz, edu ឬប្រទេសដូចជា uk, nl, de, be, au, ca, ។ល។ សូមពិនិត្យមើលពីរូបភាពខាងក្រោមនេះ



នៅឯផ្នែកខាងលើបំផុតនៃ DNS hierarchy គឺមាន 13 root name servers ដែលមានព័ត៌មានអំពី nameserver សម្រាប់ top level domain extensions។

ឧទាហរណ៍ nameserver សម្រាប់ .com និងមានព័ត៌មាននៅលើ networklessons.com ប៉ុន្តែវាមិនដឹងថាអ្វីមួយទាក់ទងជាមួយ networklessons.org នោះទេ។ វានិងធ្វើ query មកកាន់ name server ដែលទទួលខុសត្រូវសម្រាប់ org domain extension ដើម្បីទទួលបានចំលើយនោះទេ។

ខាងក្រោមនៃ top level domain extensions មាន second level domains។ នេះជាគឺជាកន្លែងដែល domain names ដូចជា networklessons, Cisco, Microsoft, ។ល។ ស្ថិតនៅទីនោះ។

បើរាប់ចុះក្រោមមកទៀតនៃ Tree អ្នកអាចរកឃើញពី hostnames ឬ subdomains។ ឧទាហរណ៍vps.networklessons.com ដែល hostname នៃ VPS (virtual private server) ជាអ្នកដំណើរការនូវ website។ ឧទាហរណ៍នៃ subdomain មួយទៀតគឺ tools.cisco.com ដែល vps.tools.cisco.com អាចជា hostname នៃ server មួយនៅក្នុង subdomain នោះ។

រវាង DNS record នីមួយៗយើងប្រើសញ្ញា . ហើយយើងក៏មានប្រើសញ្ញា . នេះសម្រាប់ root domain ដែរ។ សូមពិនិត្យមើលពីឧទាហរណ៍ពីខាងក្រោមនេះ:

vps.networklessons.com.

vps.networklessons.com

៦-៩-ការណែនាំឲ្យស្គាល់ Ethernet

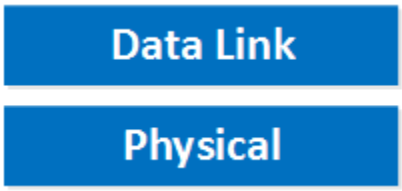
នៅលើ Local area networks(LAN) របស់យើង។ យើងប្រើតែEthernetប៉ុណ្ណោះគ្មានអ្វីក្រៅពីនោះទេ។

តើ LAN ជាអ្វី?

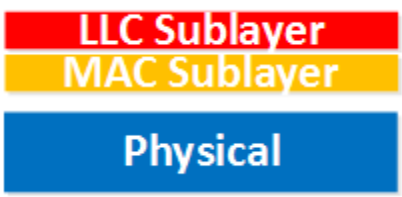
វាគឺជាប្រភេទនៃ ណេតវើកដែលគេបង្កើតឡើងសម្រាប់ភ្ជាប់នៅក្នុងអាគារឬការិយាល័យជិតៗគ្នា។

បើអ្នកមានការភ្ជាប់មកកាន់ ISP ឬប្រើ Leased-line ដើម្បីភ្ជាប់មកកាន់ ណេតវើកនៅឯការិយាល័យកណ្តាលមកកាន់សាខាមួយ ណេតវើកប្រភេទនោះគឺជា WAN (Wide Area Network) ។

Ethernet គឺជា Protocol ដែលយើងប្រើវានៅលើ LAN របស់យើង។ តើ Layer ណានៃ OSI Model ដែលអ្នកគិតថាទាក់ទងជាមួយ Ethernet? បើអ្នកកំពុងគិតថាវា “Data link” layer អ្នកពិតជាត្រឹមត្រូវ ប៉ុន្តែវាក៏ពិណ៌នាអំពី Physical layer ផងដែរ។



Ethernet ពិពណ៌នាអំពី Data link layer ប៉ុន្តែវាបានចែកចេញជាពីរផ្នែកគឺ



ដូច្នេះមាន Sublayers មានឈ្មោះថា “LLC” ដែលមានពាក្យពេញថា Logical Link Control និង “MAC” មានពាក្យពេញថា “Media Access Control” ។

logical link control layer ធ្វើការដូចជាកែ Error ជាដើម។ យើងមិនសូវចាប់អារម្មណ៍ខ្លាំងទៅលើវា នោះទេសព្វថ្ងៃនេះពីព្រោះថាយើងប្រើ TCP ដែលធ្វើការកែ Error នៅឯ Transport layer ។ ត្រូវចាំថា Ethernet ត្រូវបានបង្កើតឡើងអស់រយៈពេលយូរយោណាស់មកហើយហើយគេប្រើប្រាស់វាជាមួយ Proto -cols ជាច្រើនក្រៅពី IP មានដូចជា IPX, AppleTalk, Novell ។ល។

MAC sublayer ត្រូវបានគេចាប់អារម្មណ៍។ គ្រប់ឧបករណ៍ទាំងអស់នៅលើ LAN ត្រូវតែមាន MAC address តែមួយគត់សម្រាប់សម្គាល់ឲ្យវា។ វាដូចទៅនិង IP address ដែលស្ថិតនៅក្នុង network layer(layer 3)។ ចំណែកឯ MAC address វិញស្ថិតនៅក្នុងdata link layer (layer2)។ ចំណុចសំខាន់មួយទៀតដែល MAC sublayer ធ្វើការគឺវាមាន Channel សម្រាប់ប្រើ។ ដូច្នេះវាអាចធ្វើឲ្យកុំព្យូទ័រដែលភ្ជាប់ខ្សែជាមួយគ្នាអាចប្រើប្រាស់បានហើយអាចចែករំលែកបាន។

តើអ្នកចង់ចាំចំពោះខ្សែ ណេតវើកទាំងនេះទេ ?

គ្រប់កុំព្យូទ័រទាំងអស់នៅក្នុង ណេតវើកត្រូវបានភ្ជាប់ជាមួយខ្សែ coax តែមួយហើយអាចប្រើប្រាស់ ណេតវើកួរមគ្នាបាន។ ណេតវើកបែបនេះគឺជា half-duplex មានន័យថាមានតែកុំព្យូទ័រតែមួយគត់អាចបញ្ជូនបានចំពោះទិន្នន័យហើយកុំព្យូទ័រផ្សេងទៀតត្រូវតែរងចាំ។

សព្វថ្ងៃនេះយើងមាន full-duplex ដែលគ្រប់ឧបករណ៍ទាំងអស់អាចបញ្ជូននិងទទួលនៅពេលព្រមគ្នាបាន។ តើមានអ្វីកើតមានឡើងបើកុំព្យូទ័រពីរចាប់ផ្តើមបញ្ជូនទិន្នន័យនៅពេលព្រមគ្នា ?

វាពិតជាមាន Collision កើតមានឡើង។ អ្នកប្រហែលជាបានដឹងពីឧបករណ៍មួយគឺ HUB ។

វាជាឧបករណ៍មួយដែលគេប្រើសម្រាប់ star topology ណេតវើក។ បញ្ហារបស់ HUB គឺវាជា ឧបករណ៍អេឡិចត្រូនិកដែលអាចបញ្ជូនបន្តបាន។ បើអ្នកប្រើវាសម្រាប់ ណេតវើករបស់អ្នក វាដំណើរការជា half-duplex មានន័យថាវាអាចកើតមានឡើងនូវ Collision ។

Hub មិនដូចទៅនិង Switch នោះទេហើយវាមិនដូចទៅនិង “hub switch” ដែរ។ hubគឺជា Repeater ដ៏សាមញ្ញមួយដែលអាចអំណើរការនៅក្នុង Layer 1 នៃ OSI model រីឯ Switch វិញដំណើរការនៅក្នុង Layer 2 ។ វាជាឧបករណ៍វៃឆ្លាតដែលអាចដឹងពី MAC address ហើយភ្ជាប់ទៅកាន់ interface បានត្រឹមត្រូវ។

ការតេស្តទៅលើខ្សែ Network

ក្រោយពីតខ្សែ UTP រួចមកដើម្បីឲ្យដឹងច្បាស់ថាតើការតរបស់យើងបានត្រឹមត្រូវតាមលក្ខណៈបច្ចេកទេសឬទេ។ គេត្រូវប្រើឧបករណ៍មួយដើម្បីតេស្តទៅលើវាគឺ MICRO MAPPER។ វាជាឧបករណ៍ដែលអាចឲ្យអ្នកអាជីពណេតវើកអាចធ្វើងផ្តាត់បានយ៉ាងលឿននិងងាយដើម្បីដឹងពីសុច្ចរិតភាពនៃ Ethernet twisted pair cables។ តាមលំដាប់ MICRO MAPPER និងតេស្តទៅលើខ្សែដែលបានតជាគូសេម្រាប់ opens, shorts, reversed, crossed និង split pairs។ ក្រោយពីភ្ជាប់ហើយយើងចុចលើប៊ូតុង TEST ហើយ MICRO MAPPER និង Scan ជាស្វ័យប្រវត្តទៅលើកំហុសដែលមានស្រាប់នៅក្នុងខ្សែ។

MicroMapper Controls and LEDs

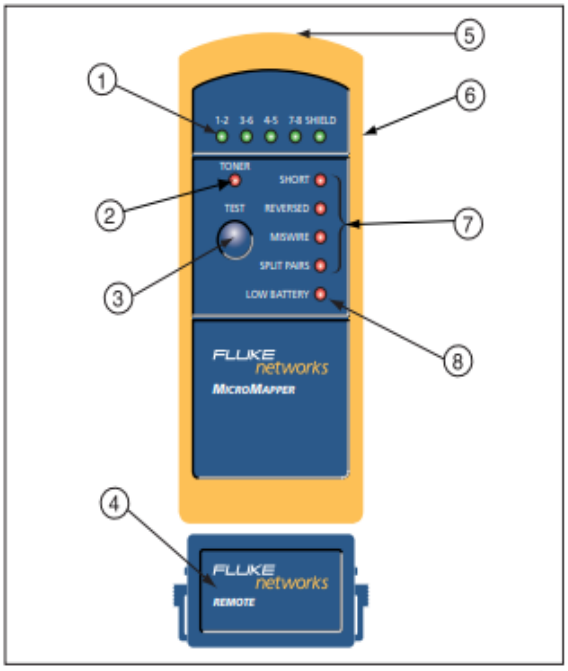
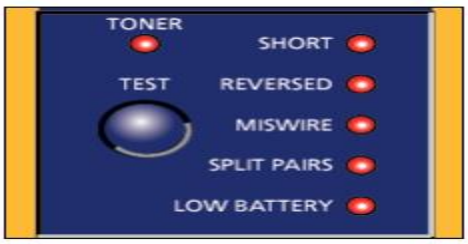


Table 1. MicroMapper Controls and LEDs

Item Number	Description
①	Pair and Shield Indicator LEDs
②	Toner LED
③	TEST button
④	REMOTE Adapter with RJ45 Jack
⑤	RJ45 Jack
⑥	Off/Cable/Toner Switch
⑦	Fault LEDs
⑧	Low Battery LED

រូបភាព១-បង្ហាញពី MICROMAPPER ណេតវើកCable Tester

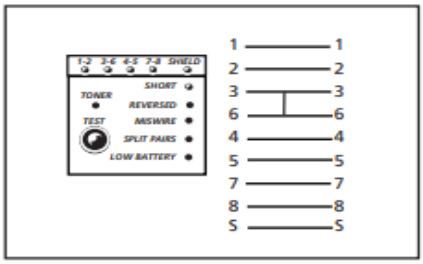


រូបភាព២-បង្ហាញពីអេក្រង់នៃ MICROMAPPER

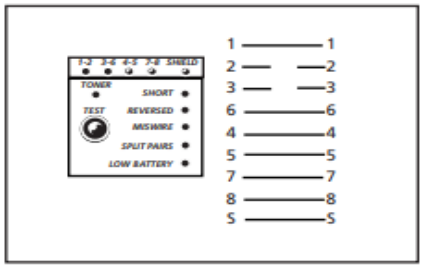
កំហុសដែលអាចកើតមានឡើងក្នុងខ្សែ ណេតវើកក្រោយពីតរួច

Fault Status

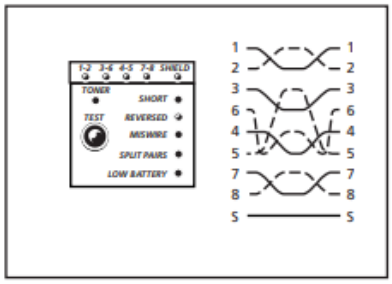
SHORT



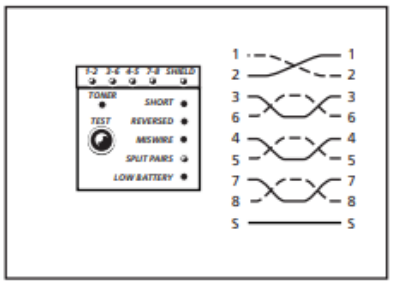
OPEN



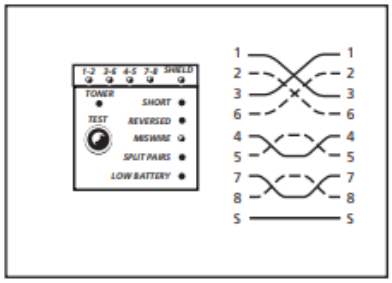
REVERSED



SPLIT PAIR



MIS-WIRE



ចំណាំ: ខ្សែ ណេតវើកដែលតបានល្អមិនជួបនូវបញ្ហាខាងលើនេះទេដូចជា OPEN SHORT REVISED SPIT និង MIS WIRE នោះទេ។

ត្រឡប់ទៅ MAC sublayer វិញ បើអ្នកកំពុងដំណើការ ណេតវើកជា half-duplex ។ បើមាន Collision កើតមានឡើងនៅលើ ណេតវើកវិញ នោះវាមានដំណោះស្រាយ។ វាមាន Protocol មួយមានឈ្មោះថា CSMA/CD:

CS = Carrier Sense

MA = Multi Access

CD = Collision Detection

- Carrier sense មានន័យថា “ស្តាប់” ទៅលើខ្សែដើម្បីឲ្យដឹងថាមានអ្វីកំពុងកើតមានឡើង។ ម៉្យាងវិញទៀត បើមានកុំព្យូទ័រមួយទៀតកំពុងបញ្ជូនទិន្នន័យ វាត្រូវតែស្តាប់។
- Multi access មានន័យថាអ្នកទាំងអស់គ្នាអាចប្រើប្រាស់ចំពោះខ្សែ ណោតវើកប៉ុន្តែត្រូវតែដឹងច្បាស់ថាគ្មានកុំព្យូទ័រណាមួយផ្សេងទៀតកំពុងបញ្ជូនទិន្នន័យនៅពេលនេះទេ។

នៅក្នុងករណីកុំព្យូទ័រពីរបញ្ជូនទិន្នន័យនៅពេលព្រមគ្នា វាមាន Collision កើតមានឡើង វាអាចរកឃើញ(carrier sense)។ CSMA/CD នឹងដោះស្រាយបញ្ហានេះដូចខាងក្រោម:

១-បើកុំព្យូទ័រពីរមាន Collision កើតមានឡើង វាត្រូវតែបញ្ជូននូវ Signal “jam” នៅលើខ្សែ។ ពេលនោះគ្មានកុំព្យូទ័រណាមួយនឹងធ្វើការបញ្ជូនទិន្នន័យនោះទេ។

២-កុំព្យូទ័រទាំងពីរនឹងដំណើរការនៅពេលចៃដន្យមួយ

៣-នៅពេលម៉ោងចៃដន្យនោះបានបញ្ចប់ វាធ្វើការបញ្ជូនឡើងវិញ

៦-៩-១-ARP (Address Resolution Protocol)

បើអ្នកបានសិក្សាពី OSI Model និង encapsulation/decapsulation អ្នកបានដឹងថានៅពេលដែលកុំព្យូទ័រពីរនៅលើ LAN ចង់ធ្វើការប្រាស្រ័យទាក់ទងគ្នា វាត្រូវតែ:

បង្កើត IP packet មួយដែលមាន IP address របស់ឧបករណ៍បញ្ជូននឹងឧបករណ៍ទទួលសម្រាប់នាំយកទិន្នន័យពី Application មួយ

IP packet នឹងត្រូវបាន encapsulated ទៅតាម Ethernet frame ដែលមានភ្ជាប់មកជាមួយនូវ MAC address របស់ source និង destination

ឧបករណ៍បញ្ជូននឹងដឹងថា MAC address របស់វា ប៉ុន្តែមិនបានដឹងពី MAC address របស់គោលដៅនោះទេ។ ដូច្នេះដើម្បីបានដឹងពី MAC address របស់គោលដៅ វាប្រើ ARP ។ សូមពិនិត្យមើលឧទាហរណ៍ខាងក្រោមនេះ

ក្នុងរូបភាពខាងលើ យើងមានកុំព្យូទ័រពីរគឺកុំព្យូទ័រ A និងកុំព្យូទ័រ B ហើយអ្នកបានដឹងពី IP addresses និង MAC addresses របស់វា។ យើងកំពុងប្រើកុំព្យូទ័រ A បើក command prompt ហើយវាយបញ្ជា:

C:Usersកុំព្យូទ័រA>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=15ms TTL=57

Reply from 192.168.1.2: bytes=32 time=15ms TTL=57

Reply from 192.168.1.2: bytes=32 time=14ms TTL=57

Reply from 192.168.1.2: bytes=32 time=17ms TTL=57

Ping statistics for 192.168.1.2:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 14ms, Maximum = 17ms, Average = 15ms

អ្នកក៏បានដឹងពី OSI Model ហើយក៏បានដឹងផងដែរចំពោះ Layers ទាំងអស់។ ចំពោះពាក្យបញ្ជា Ping ប្រើ ICMP protocol ហើយ IP ប្រើ Layer 3 ។ IP Packet របស់យើងមាន Source IP address 192.168.1.1 ហើយ IP address របស់គោលដៅគឺ 192.168.1.2 ។ ជំហានបន្ទាប់គឺវានឹងដាក់ IP Packet ចូលទៅក្នុង Ethernet frame ដែលនឹងកំនត់នូវ MAC address របស់ Source គឺ AAA ហើយ MAC address របស់គោលដៅគឺ BBB ។

តើកុំព្យូទ័រ A ដឹង MAC address របស់កុំព្យូទ័រ B តាមរបៀបណា? យើងបានដឹង IP address ដោយប្រើបញ្ជា។ ប៉ុន្តែគ្មានវិធីណាសម្រាប់កុំព្យូទ័រ A បានដឹងពី MAC address របស់កុំព្យូទ័រ B នោះទេ។ មាន Protocol មួយទៀតដែលយើងនឹងប្រើសម្រាប់ដោះស្រាយបញ្ហានេះឲ្យយើងគឺ ARP ។

សូមពិនិត្យមើលពីរបៀបដែលវាដំណើរការ

C:Usersកុំព្យូទ័រA>arp -a

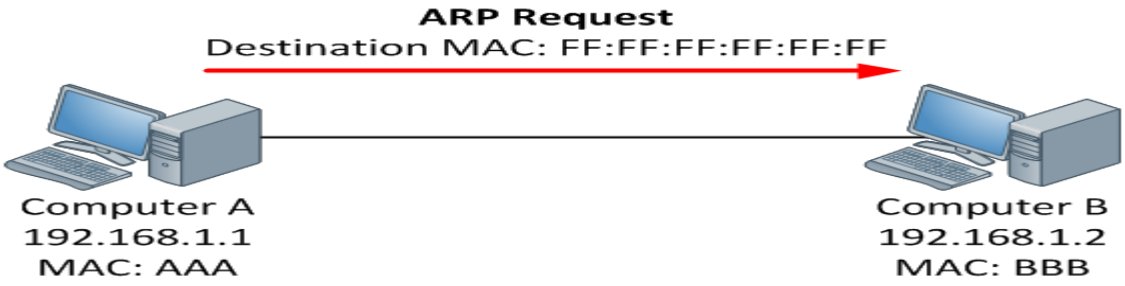
Interface: 192.168.1.1 --- 0xb

Internet Address	Physical Address	Type
192.168.1.2	00-0c-29-63-af-d0	dynamic
192.168.1 .255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.252	01-00-5e-00-00-fc	static

239.255.255.250 01-00-5e-7f-ff-fa static

255.255.255.255 ff-ff-ff-ff-ff-ff static

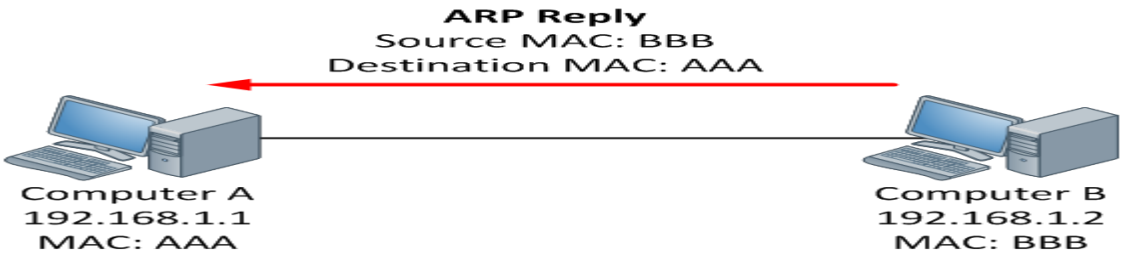
នៅក្នុងឧទាហរណ៍ខាងលើអ្នកបានឃើញពីឧទាហរណ៍នៃ ARP table នៅលើកុំព្យូទ័រ A។ ដូចអ្នកបានដឹងស្រាប់កុំព្យូទ័របានដឹងពី IP address គឺ 192.168.1.2 ត្រូវបានកំនត់ឲ្យត្រូវជាមួយ MAC address 00:0C:29:63:AF:D0។



នៅក្នុងឧទាហរណ៍នេះយើងមានកុំព្យូទ័រពីរហើយអ្នកអាចឃើញនូវ IP address និង MAC address របស់វា។ យើងកំពុងប្រើកុំព្យូទ័រ A ហើយចង់បញ្ជូននូវបញ្ជា ping មួយទៅឲ្យកុំព្យូទ័រ B។ ARP table គឺទទេ។ ដូច្នេះយើងមិនអាចដឹងពី MAC address របស់កុំព្យូទ័រ B បាននោះទេ។

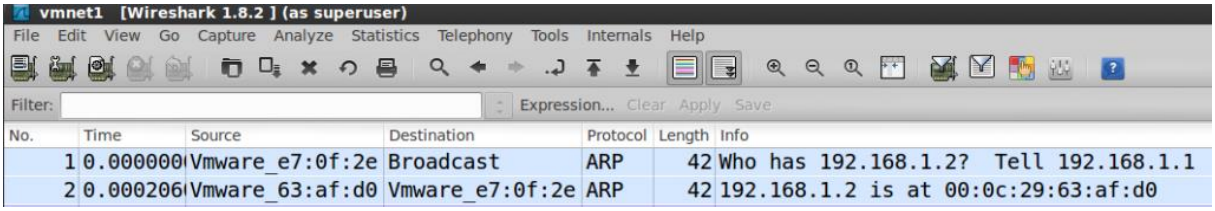
ដំបូងកុំព្យូទ័រ A នឹងបញ្ជូននូវ ARP request មួយ។ នៅក្នុង Message នោះបាននិយាយថា “Who has 192.168.1.2 and what is your MAC address?”

ដោយសារតែយើងមិនបានដឹងពី MAC address នោះវាប្រើ broadcast MAC address សម្រាប់គោលដៅគឺ (FF:FF:FF:FF:FF:FF)។ message នឹងត្រូវបានទទួលដោយគ្រប់កុំព្យូទ័រទាំងអស់នៅក្នុង ណេតវើក។



កុំព្យូទ័រ B នឹងឆ្លើយតបជាមួយ message ARP Reply ហើយនិយាយថា “នោះជាខ្ញុំ! ហើយនេះជា MAC address” របស់ខ្ញុំ។ កុំព្យូទ័រ A អាចបន្ថែម MAC address ចូលទៅក្នុង ARP table ហើយក៏ចាប់ផ្តើមបញ្ជូនបន្តទិន្នន័យទៅឲ្យកុំព្យូទ័រ B ។

បើអ្នកចង់មើលវា សូមពិនិត្យមើលក្នុង Wireshark:



ដូចក្នុងរូបខាងលើ អ្នកបានឃើញថា ARP request សម្រាប់កុំព្យូទ័រ A កំពុងស្វែងរក IP address របស់កុំព្យូទ័រ B ។ MAC address របស់ Source គឺជា MAC address របស់កុំព្យូទ័រ A ចំណែកឯ MAC address របស់កុំព្យូទ័រ B វិញគឺ “Broadcast” ។ ដូច្នេះវានិងបញ្ជូនទៅលើ ណេតវើកទាំងមូល។

តើអ្វីទៅជា Default Gateway?

នៅពេលដែល Host មួយចង់ភ្ជាប់មកកាន់គោលដៅដែលមិនស្ថិតនៅក្នុង ណេតវើកជាមួយគ្នា វាត្រូវតែប្រើ default gateway សម្រាប់ទាក់ទង។ យើងប្រើ Router មួយឬ Switch ដែលមាន Layers ច្រើន(វាអាច Route បាន) ជា default gateway ។

នៅក្នុងមេរៀននេះយើងនិងពន្យល់ពី Host មួយដឹងនៅពេលប្រើ default gateway ឬមិនប្រើ។ សូមពិនិត្យមើលពីឧទាហរណ៍ដ៏សាមញ្ញាមួយ:

ខាងលើនេះមាន Hosts ពីរភ្ជាប់គ្នាទៅវិញទៅមកជាមួយ Switch មួយ។ មាន ណេតវើកតែមួយគត់គឺ 192.168.1.0 និង subnet mask គឺ 255.255.255.0។

នៅពេលដែល Host មួយចង់ទាក់ទងបញ្ជូនអ្វីមួយទៅឲ្យ Host មួយទៀត នោះវានឹងពិនិត្យមើលថាតើគោលដៅ Host ស្ថិតនៅក្នុង ណេតវើកជាមួយគ្នាឬ ណេតវើកផ្សេងគ្នា។ នៅពេលដែល Host ជាគោលដៅស្ថិតនៅក្នុង ណេតវើកជាមួយគ្នា នោះវានិងប្រើ ARP ដើម្បីស្វែងរក MAC address របស់ Host ជាគោលដៅហើយវាក៏បញ្ជូនវ IP packet ។ តើវាពិនិត្យមើល Host ដែលជាគោលដៅស្ថិតនៅក្នុង ណេតវើកជាមួយគ្នាតាមរបៀបណា?

វាធ្វើទៅបានដោយពិនិត្យមើលទៅលើ subnet mask ។ ឧទាហរណ៍ ឧបមាថា IP address 192.168.1.1 ចង់ទាក់ទងបញ្ជូន IP packet ទៅឲ្យ IP address 192.168.1.2 ។

Source	192.168.1.1	11000000 10101000 00000001 00000001
Destination	192.168.1.2	11000000 10101000 00000001 00000010
Subnet mask	255.255.255.0	11111111 11111111 11111111 00000000

subnet mask និងប្រាប់ឲ្យដឹងថាផ្នែកនៃ IP address ពី ណេតវើកនិង Host។ Host ដែលប្រើ IP address 192.168.1.1 ដឹងថា IP address 192.168.1.2 កំពុងស្ថិតនៅក្នុង network addressជាមួយគ្នាហើយនិងដឹងថាវាអាចប្រើ ARP ដើម្បីស្វែងរក MAC Address វាបង្កើត Ethernet frame វា encapsulate IP packet ហើយក៏បញ្ជូនវាទៅឲ្យ Switch ។

ជំពូកទី៧

Subnetting

៧-១-មូលដ្ឋានគ្រឹះនៃប្រព័ន្ធគោល២

មុនពេលយើងចាប់ផ្តើមគណនាអំពី subnets និងពិភាក្សាអំពី IP address ។ ជាដំបូងត្រូវពិនិត្យមើលពីមូលដ្ឋានគ្រឹះនៃការគណនាប្រព័ន្ធគោល២ ។ យើងបានប្រើវាជាមួយប្រព័ន្ធគោល១០ដែលអាចរាប់វាពី១ទៅដល់១០។

នៅក្នុងប្រព័ន្ធគោល២មានតែលេខ០និង១ ។

0 = Off

1 = On

សូមពិនិត្យទៅលើឧទាហរណ៍ខាងក្រោមដែលបង្ហាញពីរបៀបប្រើប្រព័ន្ធគោល២ដើម្បីបង្កើតបានជាលេខគោល១០។

128	64	32	16	8	4	2	1

ខាងលើនេះមាន 8 bits ។ Bit នៅខាងឆ្វេងបំផុតហៅថា most significant bit (MSB) ពីព្រោះថា bit នេះមានតម្លៃខ្ពស់បំផុតគឺ១២៨ ។ ចំណែកឯ bit នៅខាងស្តាំបំផុតវិញហៅថា least significant bit (LSB) ពីព្រោះថាវាមានតម្លៃទាបបំផុតគឺ១ ។

៧-២-ការបំប្លែងពីប្រព័ន្ធគោល១០ទៅជាប្រព័ន្ធគោល២

ឧបមាថាយើងចង់បានលេខ 178 គោល១០ ទៅជាចង់បានជាគោល២ ។ យើងអាចធ្វើវាដោយរាប់ពីឆ្វេងហើយសាកល្បងគិតថាតើ bit នីមួយៗជាបង្កើតបានជាលេខនេះដែរឬទេ ។

128	64	32	16	8	4	2	1
1	0	1	1	0	0	1	0

128 + 32 + 16 + 2 = 178.

៧-៣-ការចែក ណេតវើកជា Subnet នៅក្នុងប្រព័ន្ធគោល២

នៅក្នុងមេរៀននេះ យើងនិងពិនិត្យមើលពីរបៀបគណនា Subnets ដោយប្រើលេខប្រព័ន្ធគោល២ ។

៧-៣-១-Class C Subnetting

សូមចាប់ផ្តើមជាមួយ class C network:

192.168.1.0 (មាន default subnet mask 255.255.255.0)

នៅក្នុងប្រព័ន្ធគោល២មានទម្រង់ជា:

192	168	1	0
11000000	10101000	00000001	00000000

ចំពោះ class C ណែតវើកhas 3 bytes ដំបូងជាផ្នែក ណែតវើកហើយមួយ byte ចុងក្រោមជា Host

Network	Network	Network	Hosts
192	168	1	0

ណែតវើកdevice ដឹងពីផ្នែកណាមួយគឺជា ណែតវើកនិង Host ដោយសារតែ Subnet mask ។ default subnet mask សម្រាប់ ណែតវើក192.168.1.0 គឺ 255.255.255.0។

ខាងក្រោមនេះគឺជាប្រព័ន្ធគោល២

IP address (decimal)	192	168	1	0
IP address (binary)	11000000	10101000	00000001	00000000
Subnet mask (decimal)	255	255	255	0
Subnet mask (binary)	11111111	11111111	11111111	00000000

ចំពោះ bit លេខ១នៅក្នុង subnet mask បង្ហាញពីផ្នែកនៃ network addressចំណែកឯ bit លេខ0បង្ហាញពីផ្នែកនៃ Host ។ ឥឡូវនេះលុបលេខគោល១0ចេញ អ្នកនឹងបានឃើញពី network addressនិង subnet mask ។

IP address	11000000	10101000	00000001	00000000
Subnet mask	11111111	11111111	11111111	00000000

ចំពោះ subnet mask ប្រាប់យើងឲ្យដឹងថា 24 bits (192.168.1) ដំបូងគឺជាផ្នែកនៃ ណែតវើកហើយ 8 bits នៅសល់ (.0)គឺជា hosts។ ចាប់ពីពេលនេះតទៅគេសម្គាល់ពី ណែតវើកមានពណ៌ក្រហម red ។ ដូច្នេះអ្នកអាចដឹងពីភាពខុសគ្នារវាង ណែតវើកនិង host bits ។ ឥឡូវនេះសរសេរ 8 bits សម្រាប់ Host:

128	64	32	16	8	4	2	1

តើតម្លៃលេខខ្ពស់បំផុតដែលអ្នកអាចបង្កើតបានជាមួយ 8 bits នោះបានលេខអ្វី? សូមកំនត់វាជាលេខ១ទាំងអស់:

128	64	32	16	8	4	2	1
1	1	1	1	1	1	1	1

128 + 64 + 32 + 16 + 8 + 4 + 2 +1 = 255

ជាមួយ 8 bits យើងអាចបង្កើតបានជា 255 ។ តើមានចំនួន Hosts 255 មែនទេក្នុង ណេតវើក ?

ចំលើយគឺទេ។ ពីព្រោះថានៅក្នុង ណេតវើកនីមួយៗមាន Addresses ចំនួន២ដែលមិនអាចប្រើបាន។ network addressទាំងនោះគឺ bits នៃ Host របស់វាគឺ0ទាំងអស់។

192	168	1	0
11000000	10101000	00000001	00000000

ចំពោះ Broadcast address គឺតម្លៃ bits នៅក្នុង host គឺ១ទាំងអស់។

192	168	1	255
11000000	10101000	00000001	11111111

ដូច្នេះ 255-2=252 ។ តើក្នុងមួយ ណេតវើកមានចំនួន Hosts 253 មែនទេ។ ចំលើយគឺមិនមែនទេ។ ចំលើយគឺ 254 ពីព្រោះថា 8 bits មានតម្លៃខ្ពស់បំផុតគឺ 256 ។ នេះមានន័យថាយើងអាចប្រើ 192.168.1.1->192.168.1.254 ។ឥឡូវនេះឧបមាថាយើងមិនចង់បាន ណេតវើកមួយដែលចំនួន Hosts ស្មើ 254 នោះទេ ប៉ុន្តែយើងចង់បានតែពីរ ណេតវើកវិញ ? តើអាចទៅរួចទេ ?

អ្វីដែលយើងត្រូវធ្វើគឺយក Class C ណេតវើកហើយបំបែកវាជាពីរ។ នេះហើយដែលគេហៅថា Subnet នោះ។ សូមពិនិត្យមើលលេខប្រព័ន្ធគោល២:

IP address (decimal)	192	168	1	0
IP address (binary)	11000000	10101000	00000001	00000000
Subnet mask (decimal)	255	255	255	0
Subnet mask (binary)	11111111	11111111	11111111	00000000

subnet mask កំនត់ពីទំហំនៃ ណេតវើក។ ដូច្នេះបើអ្នកចង់បង្កើតនូវ Subnet ច្រើន យើងត្រូវតែខ្ចី Bits ពី Host មកប្រើ។

ចំពោះ bit នីមួយៗដែលយើងខ្ចីអាចបង្កើតបានជា subnet កើនឡើងពីរដង។ បើខ្ចី 1 bit យើងអាចបង្កើតបានជា ២ subnets នៃ ណេតវើកតែមួយ។ មាន 8 bits នៅក្នុង host ។ ដូច្នេះយើងអាចខ្ចីមួយដើម្បីបង្កើតបានជា Subnet ច្រើន។ នេះមានន័យថាយើងមាន 7 bits នៅសល់សម្រាប់ host។ សូមសាកល្បងជាមួយ subnet mask :

255	255	255	128
11111111	11111111	11111111	10000000

24 bits ដំបូងដូចគ្នាហើយយើងខ្វីតែ bit ទី១ពី Octet ទី៤។ វាមានតម្លៃ ១២៨។ ដូច្នេះ Subnet mask គឺ 255.255.255.128។

ដូច្នេះតើយើងបាន Subnet ប៉ុន្មានខ្លះ? សូមពិនិត្យមើល 7 bits ដែលនៅសល់សម្រាប់ Hosts:

128	64	32	16	8	4	2	1
N/A	0	0	0	0	0	0	0

យើងមិនប្រើ bit ទី១នោះទេពីព្រោះថាគេប្រើវាសម្រាប់ network address។ តើលេខដែលមានតម្លៃខ្ពស់បំផុតនៃ 7 bits នោះមានលេខប៉ុន្មាន?

64 + 32 + 16 + 8 + 4 + 2 + 1 = 127.

ចំពោះ Class C ណោតវើកត្រូវបានគេចែកជា Subnets ចំនួន២ដែលនីមួយៗមាន ១២៨ Addresses ។ តើ Subnets ទាំងពីរនោះមានទម្រង់ដូចម្តេច?

Subnet #1

យើងចាប់ផ្តើមជាមួយ network address: 192.168.1.0 នឹងមាន subnet mask គឺ 255.255.255.128:

IP address	192	168	1	0
	11000000	10101000	00000001	00000000
Subnet mask	255	255	255	128
	11111111	11111111	11111111	10000000

network address:

network addressមាន bits នៅក្នុង Hosts ជា 0 ។ ដូច្នេះវាមានទម្រង់ជា 192.168.1.0:

192	168	1	0
11000000	10101000	00000001	00000000

IP address ទី១ ដែលអាចប្រើបាន

IP address ទី១ដែលអាចប្រើបានគឺជា IP address បន្ទាប់ពី network addressគឺ 192.168.1.1:

192	168	1	1
11000000	10101000	00000001	00000001

IP address ដែលអាចប្រើបានចុងក្រោយ

IP address ចុងក្រោយដែលអាចប្រើបានសម្រាប់ Host គឺជា IP address មុន broadcast address គឺ 192. 168. 1.126:

192	168	1	126
11000000	10101000	00000001	01111110

Broadcast address:

Broadcast address គឺជា Address ដែល bits នៅក្នុង Hosts ជា bits លេខ១ទាំងអស់។ ដូច្នេះ broadcast address we have is 192.168.1.127:

192	168	1	127
11000000	10101000	00000001	01111111

Subnet #2

subnet ទី១បញ្ចប់ដោយ 192.168.1.127 ។ ដូច្នេះយើងអាចបន្តទៅ subnet បន្ទាប់គឺ 192.168.1.128:

IP address	192	168	1	128
	11000000	10101000	00000001	10000000
Subnet mask	255	255	255	128
	11111111	11111111	11111111	10000000

network address:

network addressមាន bits នៅក្នុង hosts ជា bits 0 ទាំងអស់។ ដូច្នេះវាមានទម្រង់ជា 192.168.1.128:

192	168	1	0
11000000	10101000	00000001	10000000

IP address ទី១ដែលអាចប្រើបាន

IP address ទី១ដែលអាចប្រើបានគឺជា IP address បន្ទាប់ពី network address គឺ 192.168.1.129:

192	168	1	129
11000000	10101000	00000001	10000001

IP address ចុងក្រោយបំផុតដែលអាចប្រើបាន

IP address ចុងក្រោយបំផុតសម្រាប់ Host គឺជា IP address មុន broadcast address គឺ 192.168.1.254:

192	168	1	254
11000000	10101000	00000001	11111110

Broadcast address:

broadcast address មាន bits នៅក្នុង hosts ជាលេខ១ ។ ដូច្នេះ broadcast address មាន IP 192.168.1.255:

192	168	1	255
11000000	10101000	00000001	11111111

ណេតវើកទី១យើងបានចែកចេញជា 2 subnets ហើយយើងបានដឹងពី network addressនិង broadcast addresses ។ យើងនិងបង្ហាញពី Class C 192.168.1.0 ណេតវើកមួយទៀតដោយចែកជា ៤ subnets ។ ចំពោះ Host bits ដែលយើងអាចខ្ចីបាន យើងអាចទទួលបានចំនួន subnet ទ្វេដង ។ ដោយខ្ចី 2 bits ដូចនេះយើងទទួលបាន 4 subnets ។ តើ Subnet mask ថ្មីជាអ្វី ?

សូមពិនិត្យមើលលេខប្រព័ន្ធគោល២

255	255	255	192
11111111	11111111	11111111	11000000

ដោយគណនាចេញពីលេខប្រព័ន្ធគោល២មកជាប្រព័ន្ធគោល១០ នោះយើងទទួលបាន: 128+64 = 192 ។

Subnet mask ថ្មីគឺ 255.255.255.192។ ជាមួយ subnet mask នេះយើងទទួលបានចំនួន hosts គឺ៦ ។

Subnet #1

យើងចាប់ផ្តើមជាមួយ network address: 192.168.1.0 និង subnet mask 255.255.255.192:

IP address	192	168	1	0
	11000000	10101000	00000001	00000000
Subnet mask	255	255	255	192
	11111111	11111111	11111111	11000000

network address:

network addressគឺមាន bits នៅក្នុង Hosts ជា bit 0 ទាំងអស់។ ដូច្នេះ network addressគឺ 192.168.1.0:

192	168	1	0
11000000	10101000	00000001	00000000

IP address ដែលអាចប្រើបានទី១

IP address ដែលប្រើបានសម្រាប់ Host ទី១គឺជា IP address បន្ទាប់ពី network addressគឺ 192.168.1.1:

192	168	1	1
11000000	10101000	00000001	00000001

IP address សម្រាប់ Host ចុងក្រោយបំផុត:

IP address ចុងក្រោយបំផុតដែលប្រើសម្រាប់ Host គឺជា IP address មុន broadcast address គឺ 192.168.1.62:

192	168	1	62
11000000	10101000	00000001	00111110

Broadcast address:

Broadcast address មាន bits នៅក្នុង Hosts ជា bit 1 ។ ដូច្នេះ broadcast address គឺ 192.168.1.63:

192	168	1	63
11000000	10101000	00000001	00111111

Subnet #2

subnet ទី១បញ្ចប់ជាមួយ IP address: 192.168.1.63 ។ ដូច្នេះយើងបន្តជាមួយ Subnet បន្ទាប់គឺ 192.168.1.64:

IP address	192	168	1	64
	11000000	10101000	00000001	01000000
Subnet mask	255	255	255	192
	11111111	11111111	11111111	11000000

network address:

network addressមាន bits នៅក្នុង Hosts ជា bits 0 ទាំងអស់ ។ ដូច្នេះ network addressគឺ 192.168.1.64:

192	168	1	64
11000000	10101000	00000001	01000000

IP addressទី១ សម្រាប់ប្រើជាមួយ Host:

IP address ដែលអាចប្រើបានទី១គឺជា IP address បន្ទាប់ពី network addressគឺ 192.168.1.65:

192	168	1	65
11000000	10101000	00000001	01000001

IP address ដែលអាចប្រើបានចុងក្រោយ:

IP address ចុងក្រោយដែលអាចប្រើបានសម្រាប់ Host គឺជា IP address មុន broadcast address គឺ 192.168.1.126:

192	168	1	126
11000000	10101000	00000001	01111110

Broadcast address:

broadcast address គឺជា Address ដែលមាន bits នៅក្នុង host ជា bits 0 ទាំងអស់។ ដូច្នេះវាមាន address គឺ 192.168.1.127:

192	168	1	127
11000000	10101000	00000001	01111111

Subnet #3

subnet ទី២គឺបញ្ចប់ដោយ IP address: 192.168.1.127 ។ ដូច្នេះ Subnet បន្ទាប់គឺ 192.168. 1.128:

IP address	192	168	1	128
	11000000	10101000	00000001	10000000
Subnet mask	255	255	255	192
	11111111	11111111	11111111	11000000

network address:

network addressមាន bits នៅក្នុង Hosts ជា bit 0 ទាំងអស់។ ដូច្នេះវាមាន network addressគឺ 192.168. 1.128:

192	168	1	128
11000000	10101000	00000001	10000000

IP address ទី១ដែលប្រើបានសម្រាប់ Host

IP address ទី១ដែលប្រើបានសម្រាប់ Host គឺជា IP address បន្ទាប់ពី network address គឺ 192.168.1.129:

192	168	1	129
11000000	10101000	00000001	10000001

IP address ចុងក្រោយបំផុតដែលអាចប្រើបាន:

IP address ចុងក្រោយបង្អស់ដែលអាចប្រើបានសម្រាប់ Host គឺជា IP address មុន broadcast address គឺ192. 168.1.190:

192	168	1	190

11000000	10101000	00000001	10111110

Broadcast address:

Broadcast address មាន bits នៅក្នុង Host ជា bits 1 ទាំងអស់។ ដូច្នេះ វាមាន network addressគឺ 192.168.1.191:

192	168	1	191
11000000	10101000	00000001	01111111

Subnet #4

Subnet ទី៤បញ្ចប់ដោយ IP address: 192.168.1.191។ ដូច្នេះបន្តជាមួយ Subnet 192.168. 1.192:

IP address	192	168	1	192
	11000000	10101000	00000001	11000000
Subnet mask	255	255	255	192
	11111111	11111111	11111111	11000000

network address:

network addressមាន bits នៅក្នុង Host ជា bits 0 ទាំងអស់។ ដូច្នេះវាមាន network addressគឺ 192.168. 1.192:

192	168	1	192
11000000	10101000	00000001	11000000

IP address ដែលអាចប្រើបានទី១:

IP address ដែលអាចប្រើបានទី១គឺជា IP address បន្ទាប់ពី network addressគឺ192.168.1.193:

192	168	1	193
11000000	10101000	00000001	11000001

IP address ចុងក្រោយបង្អស់ដែលអាចប្រើបាន:

IP address ចុងក្រោយបង្អស់ដែលអាចប្រើបានគឺជា IP address មុន broadcast address គឺ 192.168.1.254:

192	168	1	254
11000000	10101000	00000001	11111110

Broadcast address:

broadcast address មាន bits នៅក្នុង Host ជា bit 1 ទាំងអស់។ ដូច្នេះ broadcast address ដែលយើងទទួលបានគឺ 192.168.1.255:

192	168	1	255
11000000	10101000	00000001	11111111

យើងទើបតែបានចែក ណេតវើក192.168.1.0 នៃ Class C ណេតវើកជា 4 subnets។ បើអ្នកយល់ពីរបៀបចែកវាហើយ អ្នកគួរតែព្យាយាមសាកល្បងវា។

នៅក្នុងមេរៀនបន្ទាប់យើងនិងដឹងពីល្បិចក្នុងការគណនា Class C, B និង Class A ជា Subnets ដោយមិនចាំបាច់ប្រើប្រព័ន្ធគោល២នោះទេ។

៧-៣-២-Class B Subnetting

យើងបានប្រើ Class C ណេតវើករួចមកហើយ។ ឥឡូវនេះសាកល្បងជាមួយ Class B ណេតវើកវិញម្តង។ គេមាន Class B ណេតវើក172.16.0.0 ដែលមាន subnet mask 255.255.0.0 ហើយបង្កើតឲ្យបានជា 2 subnets ។

IP address	172	16	0	0
	10101100	00010000	01100100	10000000
Subnet mask	255	255	0	0
	11111111	11111111	00000000	00000000

បើអ្នកចង់បង្កើតឲ្យបាន Subnets ច្រើន នោះអ្នកត្រូវតែខ្ចី Bits ពី Host bits។ គ្រប់ bits ដែលអ្នកបានខ្ចីអ្នកអាចទទួលបានចំនួននៃ subnets ទ្វេដង។ បើខ្ចី 1 bit នោះយើងទទួលបាន 2 subnets នៃ ណេតវើកដើមមួយ។

តើ Subnet mask ថ្មីជាលេខអ្វី? សូមពិនិត្យមើលពីលេខប្រព័ន្ធគោល២:

255	255	128	0
11111111	11111111	10000000	10000000

ការចែកជា Subnet ដោយប្រើប្រព័ន្ធគោល១0

នៅក្នុងមេរៀនមុន យើងបានសិក្សាពីប្រព័ន្ធគោល២ ។ ឥឡូវនេះយើងសិក្សាបន្ថែមទៀតដោយផ្តោតទៅលើប្រព័ន្ធគោល១០ ។ យើងអាចចែកជា subnets ដោយប្រើលេខប្រព័ន្ធគោល១០បាន ។

ដូចយើងបានឃើញនៅក្នុងប្រព័ន្ធគោល២ គេប្រើច្បាប់នៃស្វ័យគុណ២ ។

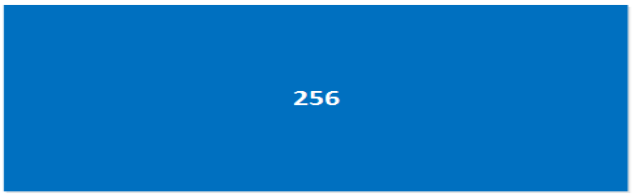
គ្រប់ bit របស់ Host ដែលអ្នកបានខ្ចីគឺជាចំនួននៃ subnets ដែលអ្នកអាចបង្កើត

គ្រប់ bit នៃ Host បង្កើនបានចំនួន Subnet ទ្វេដង

សូមពិនិត្យមើលពី ណេតវើក192.168.1.0 ដែលមាន subnet mask គឺ 255.255.255.0 ។ យើងដឹងថា subnet នេះមាន 8 bits នៅសល់ហើយ 8 bits មានតម្លៃខ្ពស់បំផុតគឺ 256 ។

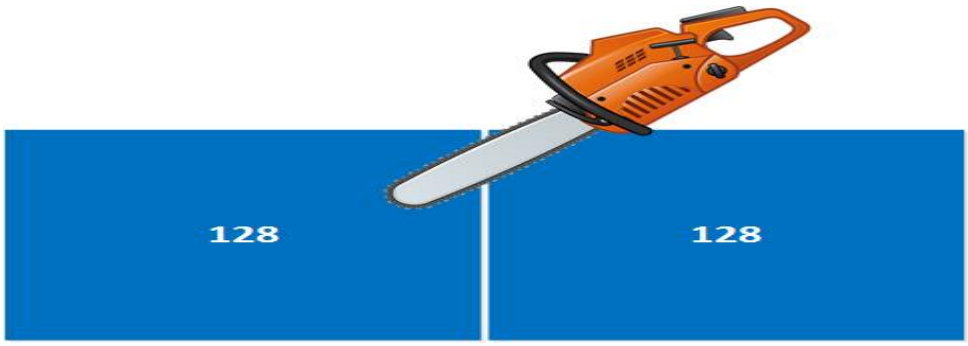
128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255.

សូមកុំភ្លេចលេខ0 ។ លេខ0ត្រូវបានប្រើ។ ដូច្នេះតម្លៃលេខខ្ពស់បំផុតដែលអ្នកអាចបង្កើតបានគឺ 256 ។



បើយើងចង់ចែក ណេតវើក192.168.1.0 ជា Subnets ។ ដូច្នេះយើងត្រូវចែកវាពីរផ្នែក ។

នៅពេលយើងចែកវាជាពីរផ្នែក យើងទទួលបាន:



ដូច្នេះយើងបង្កើតបានជា 2 subnets នៃ Class C ណេតវើក ។

តើ network addressមានលេខអ្វី ?

តើ broadcast address មានលេខអ្វី ?

តើអ្វីជា subnet mask ?

តើ IP addresses ដែលអាចប្រើបានមានអ្វីខ្លះ ?

យើងសរសេរនូវ network addressដែលវាពីរផ្នែកដែលមាន 128។ យើងចាប់ផ្តើមជាមួយ 192.168.1.0 និង subnet ទី២គឺ 192.168.1.128 (.0 – .127 = 128)។

Subnet 1:

network address: 192.168.1.0

Subnet 2:

network address: 192.168.1.128

តើ Broadcast address មានលេខអ្វីខ្លះ ?

Subnet 1:

network address: 192.168.1.0

broadcast address: 192.168.1.127

Subnet 2:

network address: 192.168.1.128

broadcast address: 192.168.1.255

តើ Subnet mask មានលេខអ្វីខ្លះ ?

ចំលើយចំពោះសំនួរនេះគឺ:

256 – “block size” = subnet mask.

ដូច្នេះនៅក្នុងឧទាហរណ៍នេះគឺ:

256 – 128 = 128.

subnet mask គឺ 255.255.255.128

តើ IP addresses ដែលអាចប្រើបានសម្រាប់ Hosts មានអ្វីខ្លះ ?

IP address ដែលអាចប្រើបានទី១បន្ទាប់ពី network address.

IP address ដែលអាចប្រើបានចុងក្រោយមុន broadcast address.

និង IP addresses នៅចន្លោះ IP addresses ដែលអាចប្រើបាន

សូមបំពេញចំពោះ Subnets ខាងក្រោម:

Subnet 1:

network address:192.168.1.0

first host: 192.168.1.1

last host: 192.168.1.126

broadcast address: 192.168.1.127

Subnet 2:

network address: 192.168.1.128

first host: 192.168.1.129

last host: 192.168.1.254

broadcast address: 192.168.1.255

នេះគឺជាវិធីសាស្ត្រដ៏លឿនមួយ។ យើងបានចែក Class C ណេតវើកជា Subnets គណនាពី network address, broadcast address និង IP addresses ដែលអាចប្រើបានចំពោះ Hosts ។

សូមសាកល្បងមួយទៀត

យើងមាន Class C ណេតវើក192.168.1.0 ប៉ុន្តែយើងនឹងចែកវាជា 4 ផ្នែក។ ដូច្នេះយើងទទួលបានជា 4 តំបន់ ។

ចូរកត់ចំលើយចំពោះសំនួរខាងក្រោម:

តើមាន network addressលេខអ្វី ?

តើមាន broadcast address លេខអ្វី ?

តើមាន Subnet mask លេខអ្វី ?

តើ IP addresses ដែលអាចប្រើបានមានលេខណាខ្លះ ?

ចំលើយចំពោះ Subnets នោះគឺយើងមានតំបន់ដែលមាន “64” :

Subnet 1:

network address: 192.168.1.0

Subnet 2:

network address: 192.168.1.64

Subnet 3:

network address: 192.168.1.128

Subnet 4:

network address: 192.168.1.192

ឥឡូវនេះយើងបានដឹងពី ណេតវើក យើងអាចសរសេរពី broadcast addresses:

Subnet 1:

network address: 192.168.1.0

broadcast address: 192.168.1.63

Subnet 2:

network address: 192.168.1.64

broadcast address: 192.168.1.127

Subnet 3:

network address: 192.168.1.128

broadcast address: 192.168.1.191

Subnet 4:

network address: 192.168.1.192

broadcast address: 192.168.1.255

តើ Subnet mask មានលេខអ្វីខ្លះ ?

256 – “block size” = subnet

256 – 64 = 192

ដូច្នេះ Subnet mask គឺ 255.255.255.192

ជំហានមួយទៀតគឺយើងត្រូវរក IP addresses ដែលអាចប្រើបានចំពោះ hosts

Subnet 1:

network address: 192.168.1.0

first host: 192.168.1.1

last host: 192.168.1.62

broadcast address: 192.168.1.63

Subnet 2:

network address: 192.168.1.64

first host: 192.168.1.65

last host: 192.168.1.126

broadcast address: 192.168.1.127

Subnet 3:

network address: 192.168.1.128

first host: 192.168.1.129

last host: 192.168.1.190

broadcast address: 192.168.1.191

Subnet 4:

network address: 192.168.1.192

first host: 192.168.1.193

last host: 192.168.1.254

broadcast address: 192.168.1.255

ដូច្នេះអ្នកបានយល់ពីក្បួនច្បាប់វាហើយ។ អ្នកអាចគណនាក្នុងប្រព័ន្ធគោល១០បានយ៉ាងលឿនហើយបន្ទាប់មកគណនានៅក្នុងប្រព័ន្ធគោល២។

តើយើងអាចអនុវត្តបានចំពោះ Class B ណែតវើកដោយប្រើវិធីសាស្ត្រនេះបានទេ ? ប្រាកដជាបាន ។

សូមពិនិត្យមើល network address172.16.0.0 និងការបង្កើត 8 subnets ។

ចូរកចំលើយចំពោះសំនួរខាងក្រោមនេះ:

តើមាន network addressលេខអ្វី ?

តើមាន broadcast address លេខអ្វី ?

តើមាន Subnet mask លេខអ្វី ?

តើ IP addresses ដែលអាចប្រើបានមានលេខណាខ្លះ ?

៧-៤-Classless InterDomain Routing (CIDR)

ចាប់តាំងពីគេបានបង្កើត IP address មនុស្សដែលមានបង្កើតវាបានគិតថាវាមានតែ Class បីប្រភេទខុសគ្នាគឺ Class A,B និង C networks ។

Class A: 255.0.0.0 (16.777.216 addresses)

Class B: 255.255.0.0 (65.536 addresses)

Class C: 255.255.255.0 (256 addresses)

Networks ទាំងនេះគេហៅថា classful networks ។

នៅពេលដែល Internet បានកើនឡើងយ៉ាងឆាប់រហ័យនៅក្នុងទសវត្សៈ៩០ ដែលជាកត្តាធ្វើឲ្យមានបញ្ហាកើតមានឡើង ។

ក្រុមហ៊ុនដ៏ធំទទួលបាន Class A networks ទាំងមូលដែលមានរាប់លាន IP addresses ។ ចំពោះក្រុមហ៊ុនតូចទទួលបាន Class B networks ដែលមានចំនួន Hosts គឺ 65.536 addresses ហើយ class C networks មាន 256 addresses ។

ជាដំណោះស្រាយចំពោះបញ្ហានេះគឺគេបានបង្កើតឡើង classless interdomain routing ។ Classless networks មានន័យថាយើងមិនប្រើ Class A,B និង C បន្តទៀតទេគឺប្រើ Subnet តាមចិត្តយើងចង់បាន ។ គេក៏ប្រើបានជំនួស 255.255.255.0 ដោយប្រើសញ្ញា /24 ។

ឧទាហរណ៍:

192.168.1.0 មាន subnet mask 255.255.255.0 ដូចគ្នាទៅនិង192.168.1.0 /24.

172.16.0.0 មាន subnet mask 255.255.0.0 ដូចគ្នាទៅនិង172.16.0.0 /16.

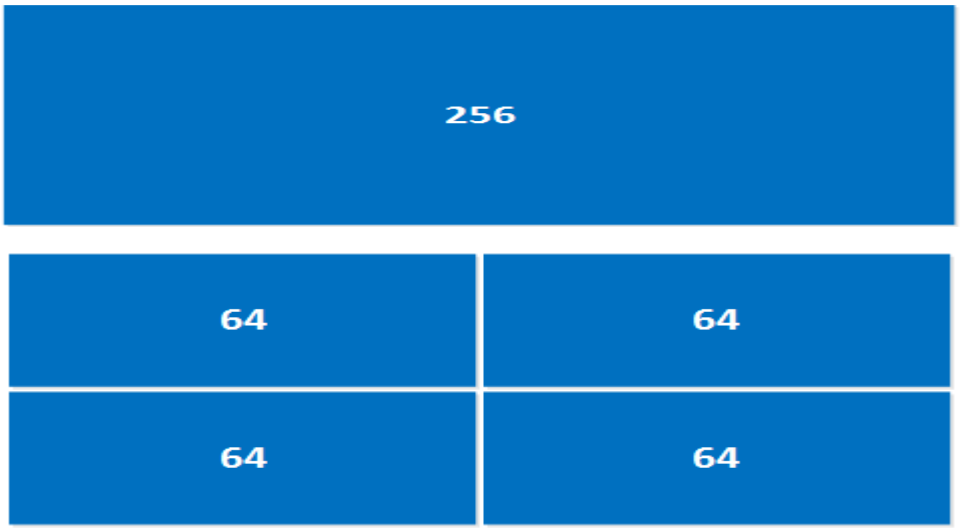
10.0.0.0 មាន subnet mask 255.0.0.0 ដូចគ្នាទៅនិង10.0.0.0 /8.

ខាងក្រោមនេះគឺជា Subnet maks និង សញ្ញាសម្គាល់នៃ CIDR:

255.0.0.0	/8
255.128.0.0	/9
255.192.0.0	/10
255.224.0.0	/11
255.240.0.0	/12
255.248.0.0	/13
255.252.0.0	/14
255.254.0.0	/15
255.255.0.0	/16
255.255.128.0	/17
255.255.192.0	/18
255.255.224.0	/19
225.225.240.0	/20
255.255.248.0	/21
255.255.252.0	/22
255.255.254.0	/23
255.255.255.0	/24
255.255.255.128	/25
255.255.255.192	/26
255.255.255.224	/27
255.255.255.240	/28
255.255.255.248	/29
255.255.255.252	/30

៧-៥-Variable Length Subnet Mask (VLSM)

នៅក្នុងមេរៀនមុនបានបង្ហាញឲ្យដឹងថា subnet មានប្រវែងថេរ។ Subnet នីមួយៗមានប្រវែងដូចគ្នា។ ឧទាហរណ៍ យើងសិក្សាទៅលើ Class C ណេតវើកដែលមាន network address192.168.1.0 ដែលបានចែកជា 4 ផ្នែក



តើនេះជាវិធីសាស្ត្រដ៏មានប្រសិទ្ធភាពក្នុងការបង្កើត subnets មែនទេ ? សូមពិនិត្យមើលពីតម្រូវការដូចខាងក្រោម:

Subnet មួយមាន 12 hosts

Subnet មួយមាន 44 hosts

Subnet មួយមាន 2 hosts (point-to-point links ដែលត្រូវការតែ 2 IP host addresses)

Subnet មួយមាន 24 hosts

បើយើងត្រូវការតែ 4 subnets នោះមិនបញ្ហានោះទេ ហើយវាមាន IP addresses ជាច្រើន។ បើយើងមានតំបន់ដែលមាន 64 សម្រាប់ subnets របស់យើងដែលយើងត្រូវការតែ 2 IP addresses នោះយើងបោះចោលនូវ 62 IP addresses ។

ឥឡូវនេះយើងចាប់អារម្មណ៍ទៅលើវាពីព្រោះថាយើងប្រើនូវ private network address(192.168.1.0) ហើយមាន Addresses ជាច្រើន។ វាជាការពិត ចុះបើប្រព័ន្ធ Internet វិញ ? យើងប្រាកដជាមិនត្រូវបោះចោលនូវ public IP addresses នោះទេ ។

ឧបមាថាយើងចង់ចែក ណេតវើក192.168.1.0 ទៅតាមតម្រូវការដូចខាងក្រោម:

Subnet មួយមាន 12 hosts

Subnet មួយមាន 44 hosts

Subnet មួយមាន 2 hosts

Subnet មួយមាន hosts

តើយើងត្រូវការនូវ Subnets ប្រភេទណាដើម្បីបំពេញចំពោះតម្រូវការ Hosts ទាំងនេះ ?

12 hosts គឺត្រូវការនូវ Subnets ដែលមានចំនួន Hosts 16

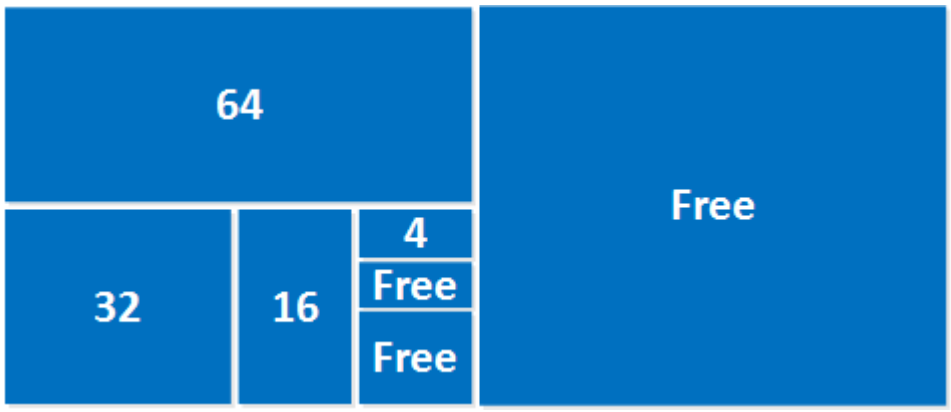
44 hosts គឺត្រូវការនូវ Subnet ដែលមានចំនួន Hosts 64

2 hosts គឺត្រូវការនូវ Subnets ដែលមានចំនួន Hosts 4

24 hosts គឺត្រូវការនូវ Subnet ដែលមានចំនួន Hosts 32

បង្កើត Subnets ទាំងនេះ។ យើងយកចំនួន Hosts 256:

យើងចែកវាជាផ្នែកដែលមានដូចខាងក្រោម:



ឥឡូវនេះយើងឆ្លើយនូវសំណួរខាងក្រោម:

តើ network addresses មានលេខអ្វីខ្លះ ?

តើ broadcast addresses មានលេខអ្វីខ្លះ ?

តើមាន Subnet mask អ្វីខ្លះ ?

តើ IP addresses ដែលអាចប្រើបានមានលេខណាខ្លះ ?

នៅពេលយើងប្រើ VLSM ត្រូវប្រាកដថាយើងចាប់ផ្តើមជាមួយ Subnet ទី១ដែលមានទំហំធំជាងគេបំផុត។ ចំលើយចំពោះសំណួរមានដូចខាងក្រោមនេះ:

Subnet 1: (size of 64)

network address: 192.168.1.0

Subnet 2: (size of 32)

network address: 192.168.1.64

Subnet 3: (size of 16)

network address: 192.168.1.96

Subnet 4: (size of 4)

network address: 192.168.1.112

Subnet 5: (this is where the free space starts)

network address: 192.168.1.116

Now we can fill in the broadcast addresses:

Subnet 1: (size of 64)

network address: 192.168.1.0

broadcast address: 192.168.1.63

Subnet 2: (size of 32)

network address: 192.168.1.64

broadcast address: 192.168.1.95

Subnet 3: (size of 16)

network address: 192.168.1.96

broadcast address: 192.168.1.111

Subnet 4: (size of 4)

network address: 192.168.1.112

broadcast address: 192.168.1.115

ដោយសារតែយើងមានទំហំនៃ Subnets ខុសគ្នា យើងត្រូវការគណនានូវ Subnet mask សម្រាប់ Subnet នីមួយៗ។ ដើម្បីរក Subnet mask យើងអាចប្រើវិធីសាស្ត្រដូចខាងក្រោម:

256 – subnet size = subnet mask

Subnet 1: 256 – 64 = 192 so the subnet mask is 255.255.255.192

Subnet 2: 256 – 32 = 224 so the subnet mask is 255.255.255.224

Subnet 3: 256 – 16 = 240 so the subnet mask is 255.255.255.240

Subnet 4: 256 – 4 = 252 so the subnet mask is 255.255.255.252

ចំណុចមួយទៀតនៅសល់គឺត្រូវបំពេញនូវ IP addresses ដែលអាចប្រើបាន:

Subnet 1: (size of 64)

network address: 192.168.1.0

first host: 192.168.1.1

last host: 192.168.1.62

broadcast address: 192.168.1.63

Subnet 2: (size of 32)

network address: 192.168.1.64

first host: 192.168.1.65

last host: 192.168.1.94

broadcast address: 192.168.1.95

Subnet 3: (size of 16)

network address: 192.168.1.96

first host: 192.168.1.97

last host: 192.168.1.110

broadcast address: 192.168.1.111

Subnet 4: (size of 4)

network address: 192.168.1.112

first host: 192.168.1.113

last host: 192.168.1.114

broadcast address: 192.168.1.115

ដូច្នេះយើងទទួលបាន Subnets ចំពោះ 192.168.1.0 /24 ដោយប្រើ VLSM ។

សូមព្យាយាមឧទាហរណ៍មួយទៀត ។ ប៉ុន្តែពេលនេះប្រើ Class B 172.16.0.0 ណេតវើកដែលមានតម្រូវការផ្សេងគ្នា:

One subnet for 340 hosts.

One subnet for 250 hosts.

One subnet for 31 hosts.

One subnet for 20 hosts.

One subnet for 8 hosts.

៧-៦-ការសង្ខេបពី Route

នៅក្នុងមេរៀននេះយើងនិងពិនិត្យមើលពីរបៀប Configure ស្តីពីការសង្ខេបនៃ Route ។ ជាដំបូងយើងនិងឃើញពីឧទាហរណ៍ដែលនិយាយអំពីរបៀបធ្វើវានៅក្នុងប្រព័ន្ធគោល២ហើយបន្ទាប់មកមើលពីរបៀបធ្វើវានៅក្នុងប្រព័ន្ធគោល១០ ។

ឧបមាថាយើងចង់បង្កើតនូវការសង្ខេបអំពី 4 networks ដូចខាងក្រោម:

192.168.0.0 / 24 subnet mask 255.255.255.0

192.168.1.0 / 24 subnet mask 255.255.255.0

192.168.2.0 / 24 subnet mask 255.255.255.0

192.168.3.0 / 24 subnet mask 255.255.255.0

ត្រូវបំប្លែង Networks ទាំងនេះជាគោល២

192.168.0.0	11000000	10101000	00000000	00000000
192.168.1.0	11000000	10101000	00000001	00000000

192.168.2.0	11000000	10101000	00000010	00000000
192.168.3.0	11000000	10101000	00000011	00000000

ឥឡូវនេះសូមពិនិត្យមើលថាតើមានប៉ុន្មាន bits ដែល Networks ទាំងនេះមានរួមគ្នា? ចំពោះ Octet ទី១និងទី២ ដូចគ្នា។ សូមពិនិត្យមើលចំពោះ Octet ទី៣:

00000000
00000001
00000010
00000011

6 bits ដំបូងនៃ octet ទី៣ ដូចគ្នា។ យើងអាចមានព័ត៌មានគ្រប់គ្រាន់សម្រាប់បង្កើត Address សង្ខេបគឺ

8 + 8 + 6 = 22 bits

ដូច្នេះ Address សង្ខេបគឺ 192.168.0.0/22 (subnet mask 255.255.252.0) ។

ឥឡូវនេះយើងបានដឹងពីរបៀបគណនាវានៅក្នុងប្រព័ន្ធគោល២។ សូមគណនាវានៅក្នុងប្រព័ន្ធគោល១០វិញ។ ខាងក្រោមនេះគឺជាល្បិចនៃការគណនាស្តីអំពីការសង្ខេប។

ដូចអ្នកបានដឹងថាយើងមាន 4 networks ឬនៅពេលអ្នកនិយាយអំពីតំបន់ វាជាតំបន់ដែលមាន 4។ នេះគឺជារូបមន្តដែលអ្នកអាចប្រើបាន:

256 – number of networks = subnet mask for summary address.

ឧទាហរណ៍

256 – 4 networks = 252

subnet mask គឺ 255.255.252.0

មានវិធីមួយទៀតដើម្បីរកវាដោយប្រើសញ្ញា CIDR។ អ្នកដឹងថា /24គឺជា Addressដែលមាន 256។

ការប្រើ /23 មានន័យថា 2 x 256 ហើយ /22 មានន័យថា 4 x 256 ។សូមពិនិត្យមើលឧទាហរណ៍មួយទៀត។ ឧបមាថាយើងចង់សង្ខេបពី Networks ខាងក្រោម:

172.16.0.0 / 16 subnet mask 255.255.0.0

172.17.0.0 / 16 subnet mask 255.255.0.0

172.18.0.0 / 16 subnet mask 255.255.0.0

172.19.0.0 / 16 subnet mask 255.255.0.0

172.20.0.0 / 16 subnet mask 255.255.0.0

172.21.0.0 / 16 subnet mask 255.255.0.0

172.22.0.0 / 16 subnet mask 255.255.0.0

172.23.0.0 / 16 subnet mask 255.255.0.0

សូមពិនិត្យមើលនៅក្នុងប្រព័ន្ធគោល២។ យើងនិងសរសេរចំពោះ octet ទី២ពីព្រោះថា Octet ទី១មានដូចគ្នាចំពោះ network addresses ទាំងអស់។

16	00010000
17	00010001
18	00010010
19	00010011
20	00010100
21	00010101
22	00010110
23	00010111

ចំពោះ 5bits ដូចគ្នាទាំងអស់។ Octet ទី១មាន 8 bits ដូច្នេះយើងទទួលបាន 8 + 5 = 13 bits។

Address សង្ខេបគឺ 172.16.0.0 /13 (subnet mask will be 255.248.0.0) ។

ការគណនាចំពោះគោល២ វាហាក់បីដូចជាយឺតបន្តិច។ សូមមើលពីការគណនាក្នុងគោល១០វិញ។

256 – number of networks = subnet mask for summary address

ដូច្នេះ 256 – 8 = 248។ subnet mask គឺ 255.248.0.0 ។

យើងអាចរកវាដោយប្រើសញ្ញា CIDR:

172.16.0.0 /16 is one network.

172.16.0.0 /15 are two networks.

172.16.0.0 /14 are four networks.

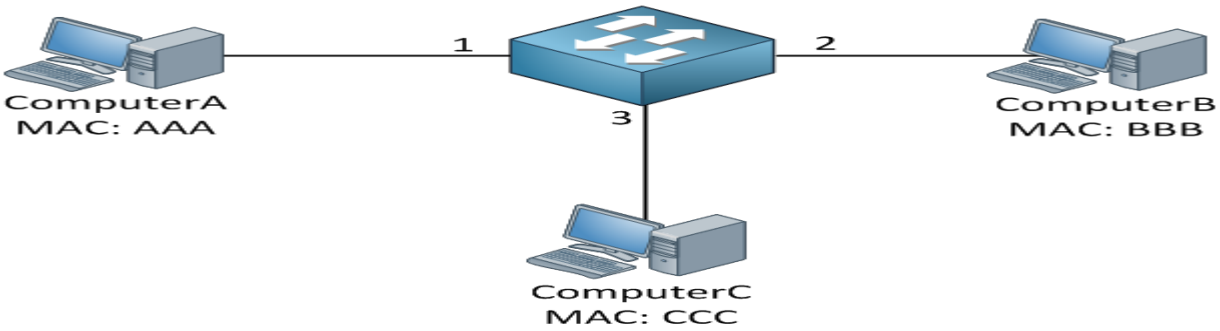
172.16.0.0 /13 are eight networks ។ វាលឿនជាងការប្រើប្រព័ន្ធគោល២

ជំពូកទី៨

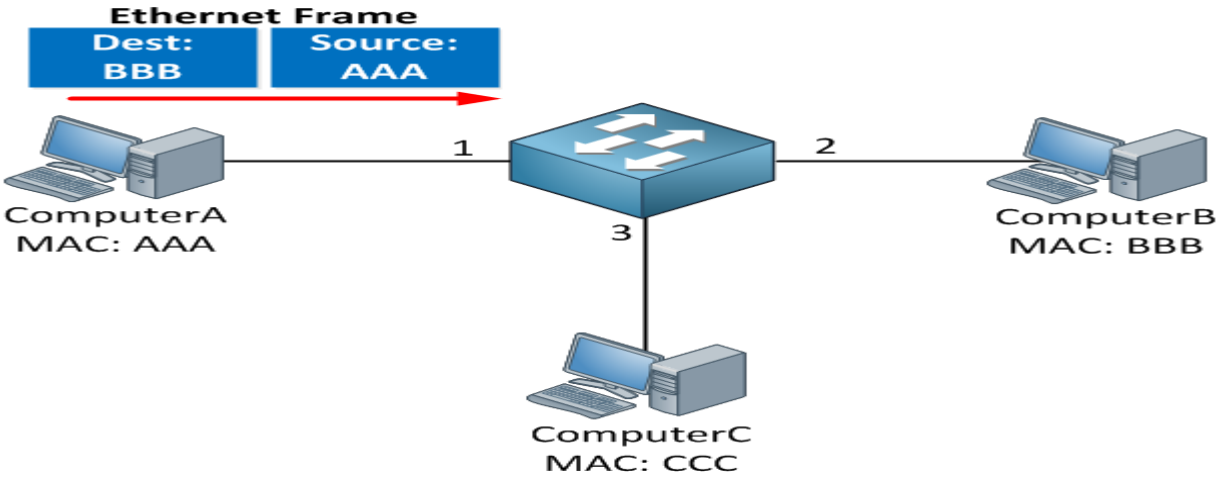
Switching

៨-១-របៀបដែល Switch ស្គាល់ MAC Addresses

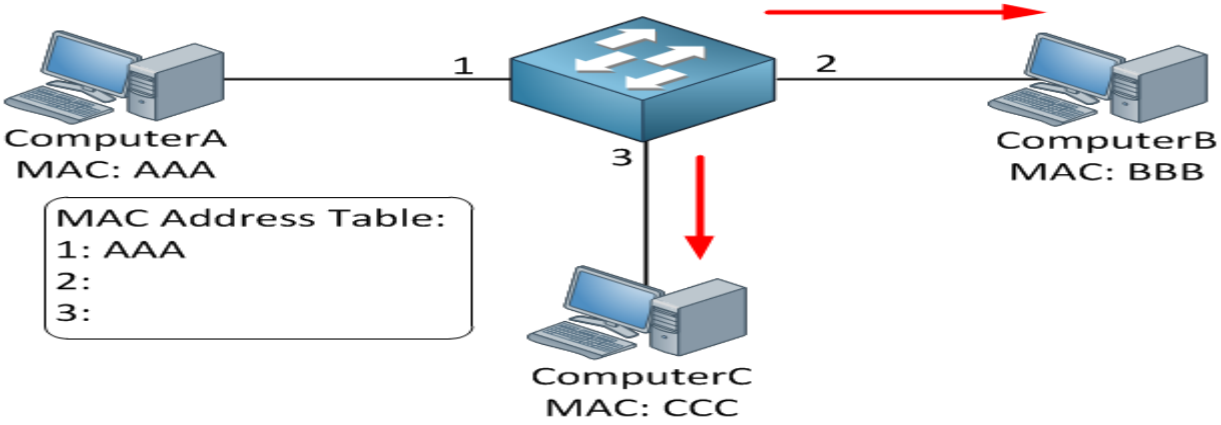
Switch គឺជាឧបករណ៍វៃឆ្លាតមួយហើយដំណើរការនៅលើ Layer 2 នៃ OSI Model ។ នៅក្នុងជំពូកនេះគឺយើងសិក្សាពីរបៀបដែល Switch មួយស្គាល់ពី MAC addresses ។ សូមពិនិត្យមើលពីឧទាហរណ៍នៃកុំព្យូទ័រ៣ភ្ជាប់ជាមួយ Switch មួយ ។



មាន Switch មួយនៅកណ្តាលហើយមានកុំព្យូទ័រ៣ ។ គ្រប់កុំព្យូទ័រទាំងអស់ត្រូវបានភ្ជាប់ជាមួយ Switch មាន MAC address របស់វា ។ Switch មាន MAC addeesses table មួយហើយវានិងស្គាល់អំពី MAC addresses ទាំងនោះដែលមាននៅក្នុង Network ។ ឧបមាថាបញ្ជូនអ្វីមួយពីកុំព្យូទ័រ A ទៅកុំព្យូទ័រ B:

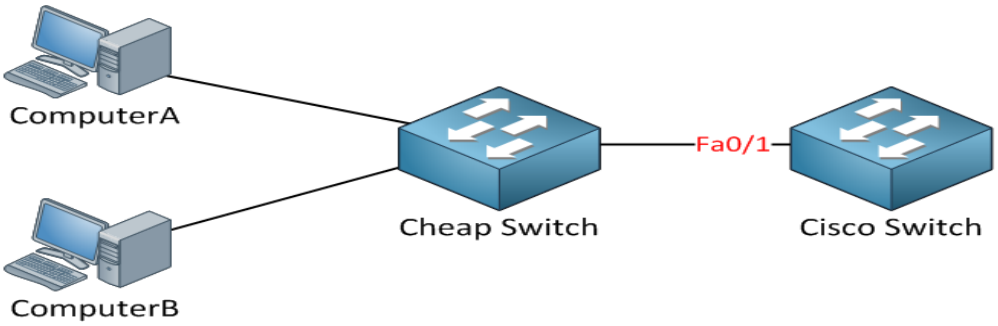


កុំព្យូទ័រ A កំពុងបញ្ជូនទិន្នន័យទៅឲ្យកុំព្យូទ័រB។ ដូច្នេះវានិងបង្កើតនូវ Ethernet frame មួយដែលមាន MAC address (AAA) ជាប្រភពបញ្ជូននិង destination MAC address (BBB) ។ Switch មាន MAC address tableមួយ ។



៨-២-របៀប configure port-security នៅលើ Cisco Switch

តាមលំនាំដើម គ្មានការកំណត់ចំនួននៃ MAC addresses ចំពោះ Switch នោះទេហើយគ្រប់ MAC addresses ទាំងអស់ត្រូវបានអនុញ្ញាតឲ្យប្រើជាមួយវាបាន។ បើអ្នកចង់ប្តូរលក្ខណៈនេះជាមួយ port-security។ សូមពិនិត្យមើលចំពោះស្ថានការណ៍ដូចខាងក្រោម:



នៅក្នុង Topology ខាងលើ គេបានភ្ជាប់វាជាមួយ Cheap Switch ទៅនិង Cisco Switch តាម FastEthernet 0/1 interface ។ ជាលទ្ធផល Cisco Switch បានស្គាល់ពី MAC Addresss របស់កុំព្យូទ័រ A និងកុំព្យូទ័រ B នៅលើ FastEthernet 0/1 interface របស់វា។

តាមការពិត យើងមិនចង់ឲ្យអ្នកណាយក Switch ពីខាងក្រៅមកភ្ជាប់ជាមួយ Switch របស់យើងនោះទេ។ យើងបានការពារមិនឲ្យវាអាចភ្ជាប់បាននោះទេ។ ចំពោះ Port របស់ Switch ត្រូវស្ថិតនៅក្នុង Access mode ដើម្បីអាចបើក Port security status បាន។

```
Switch(config)#interface fa0/1

Switch(config-if)#switchport port-security

Switch(config-if)#switchport port-security maximum 1
```

ការប្រើ Switch port-security គឺអាចបើក Port-security ។ គេបាន Configure port-security ដូច្នេះមានតែ MAC address តែមួយគត់ដែលត្រូវបានអនុញ្ញាត។ នៅពេលដែល Switch បានឃើញ MAC address ផ្សេងទៀតនៅលើ Interface វានិងបំពានទៅលើការកំនត់នោះ។

ក្រៅពីកំនត់ចំនួនអតិបរមានៃ MAC addresses យើងក៏អាចប្រើ Port security ដើម្បីប្រោះ MAC address បានដែរ។ អ្នកអាចប្រើបានតែ MAC addresses ដែលមានការអនុញ្ញាតតែប៉ុណ្ណោះ។ នៅក្នុងឧទាហរណ៍នេះគេបាន Configure port security ចំពោះ MAC address aaaa.bbbb.cccc ។

Switch(config)#interface fa0/1

Switch(config-if)#switchport port-security mac-address aaaa.bbbb.cccc

គេប្រើ swtichport port-secuiryt mac-address ដើម្បីកំណត់ MAC address ដែលអ្នកចង់អនុញ្ញាតឲ្យប្រើ។ ឥឡូវនេះវានិងបង្កើតបាននូវចារចរណ៍ដែលបណ្តាលឲ្យមានការបំពានមួយ។

C:\Documents and Settings\កុំព្យូទ័រA>ping 1.2.3.4

SwitchA#

%PM-4-ERR_DISABLE: psecure-violation error detected on Fa0/1, putting Fa0/1 in err-disable state

%PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred, caused by MAC address 0090.cc0e.5023 on port FastEthernet0/1.

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

%LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to down

យើងបានបំពានទៅលើសុវត្ថិភាព។ ជាលទ្ធផលគឺវាបង្ហាញ Port Error ដែលមានជា err-disable state ។ ដូចដែលអ្នកបានដឹងថាវាត្រូវបានបិទចោល។

សូមពិនិត្យមើលពី port-security ឲ្យបានដិតដល់ដូចខាងក្រោម:

Switch#show port-security interface fa0/1

Port Security : Enabled

Port Status : Secure-shutdown

Violation Mode : Shutdown

Aging Time : 0 mins

Aging Type : Absolute

SecureStatic Address Aging : Disabled

Maximum MAC Addresses : 1

Total MAC Addresses : 1

Configured MAC Addresses : 1

Sticky MAC Addresses : 0

Last Source Address:Vlan : 0090.cc0e.5023:1

Security Violation Count : 1

នេះគឺជាបញ្ហាដ៏សំខាន់ដើម្បីត្រួតពិនិត្យទៅលើការ configure ចំពោះ Port Security។ ការប្រើ show port-security interface គឺសម្រាប់មើលបីព័ត៌មានពិស្តារនៃ Interface មួយ។ អ្នកអាចឃើញពីការរំលោភបំពានត្រូវបានបិទចោលហើយការបំពានចុងក្រោយត្រូវបំពានដោយ MAC address 0090. cc0e. 5023 (កុំព្យូទ័រA)។ រយៈពេលនៃអាយុកាលគឺ 0 នាទីមានន័យថាវាស្ថិតនៅក្នុង err-disable ជារៀងរហូត។

Switch#show interfaces fa0/1

FastEthernet0/1 is down, line protocol is down (err-disabled)

ការបិទចោលចំពោះ Interface ក្រោយពីមានការបំពានទៅលើសុវត្ថិភាពគឺជាគំនិតល្អ ប៉ុន្តែបញ្ហាគឺ interface ស្ថិតនៅក្នុងសភាព stay in err-disable state។

ដូច្នេះគឺត្រូវការអ្នកជំនួយបើកវាឲ្យដំណើរការវិញ។

Switch(config)#interface fa0/1

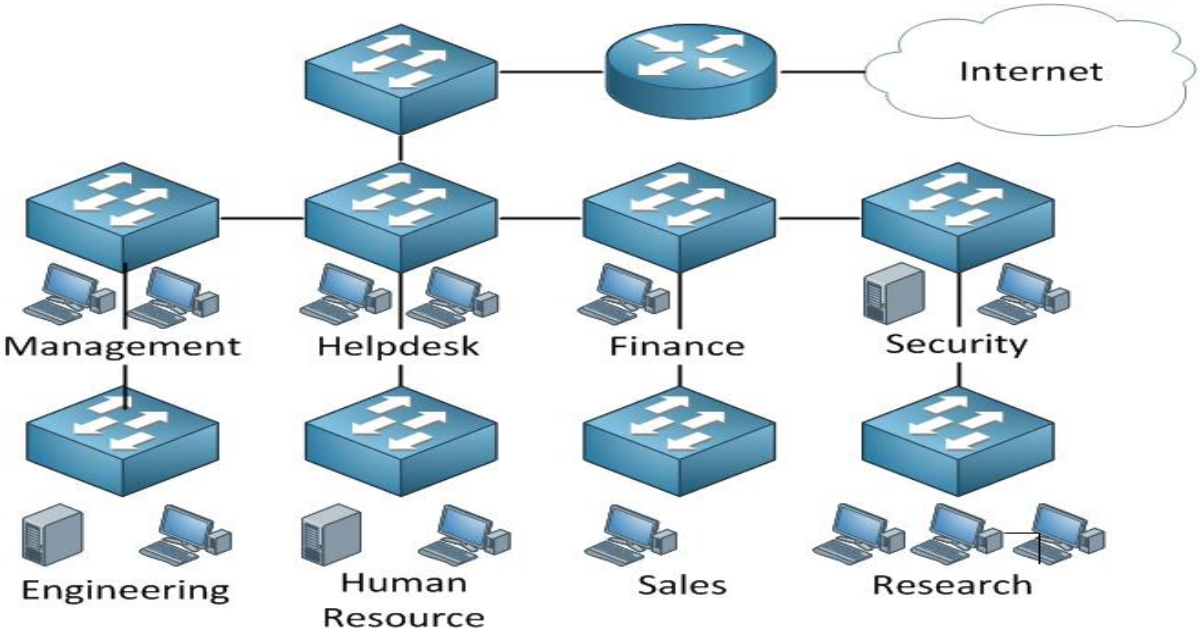
Switch(config-if)#shutdown

Switch(config-if)#no shutdown

៨-៣-សេចក្តីផ្តើមចំពោះ VLANs

នៅក្នុងមេរៀននេះពិនិត្យមើលទៅលើ VLANs (Virtual LANs) ហើយយើងនិងសិក្សាហេតុអ្វីបានជាត្រូវការវា។

ជាបឋមសូមពិនិត្យមើលពីរូបភាពនៃ ណេតវើកខាងក្រោម:



នៅក្នុងរូបភាពយើងមានដេបាតឺម៉ង់ជាច្រើនហើយដេប៉ាតឺម៉ង់នីមួយៗមាន Switch មួយផ្ទាល់ខ្លួន។ អ្នកប្រើប្រាស់ត្រូវបានចែកជាក្រុមតាមដេប៉ាតឺម៉ង់ហើយភ្ជាប់ទៅនិង Switch របស់គេ។ តើអ្នកគិតឃើញថាដូចម្តេចចំពោះវា? តើវាជាការ Design ណេតវើកមួយដ៏ល្អ? បើអ្នកមិនប្រាកដទេ សូមស្តាប់ការពន្យល់ខាងក្រោមនេះ:

តើមានអ្វីកើតមានឡើងនៅពេលដែលកុំព្យូទ័រមួយភ្ជាប់ទៅ Research Switch បញ្ជូន broadcast ដូចជា ARP request?

តើមានអ្វីកើតមានឡើងនៅពេលដែល Helpdesk switch បរាជ័យ?

តើអ្នកប្រើប្រាស់នៅឯ Human ធនធាន switch ទទួលបានការភ្ជាប់ ណេតវើកបានលឿន?

តើយើងអនុវត្តសុវត្ថិភាពចំពោះ ណេតវើកនេះតាមរបៀបណា?

សូមពន្យល់ថាហេតុអ្វីបានជាការ Design ណេតវើកមួយមិនល្អ? បើកុំព្យូទ័រណាមួយបញ្ជូន broadcast មួយទៅ តើមានអ្វីកើតមានឡើងចំពោះ Switches? វាបញ្ជូនទៅឲ្យទាំងអស់(flooding)។ មានន័យថា broadcast frame តែមួយនិងត្រូវបានបញ្ជូនទៅឲ្យ ណេតវើកទាំងមូល។ បញ្ហានេះក៏អាចកើតមានឡើងផងដែរនៅពេលដែល Switch មិនស្គាល់ពី MAC address ពិតប្រាកដ។

បើ helpdesk Switch បរាជ័យ មានន័យថាអ្នកប្រើប្រាស់ពី Human ធនធាន ត្រូវបានផ្តាច់ចេញពីគេ ហើយមិនអាចចូលប្រើជាមួយដេប៉ាតឺម៉ង់ផ្សេងបានឬ internet។ អ្នកប្រើប្រាស់ទាំងអស់គ្នាត្រូវតែឆ្លងកាត់តាម Helpdesk switch ដើម្បីអាចប្រើ Internet បាន។ មានន័យថាយើងបានប្រើ Bandwidth រួមគ្នា ។

តើសុវត្ថិភាពយ៉ាងដូចម្តេចដែរ? យើងគួរប្រើ Port-security ហើយច្រោះទៅលើ MAC addresses ប៉ុន្តែត្រូវចាំថាវាមិនស៊ូវមានសុវត្ថិភាពនោះទេពីព្រោះថា MAC addresses គេអាចប្តូរបានយ៉ាងងាយ។ គេប្រើ VLANs ជាដំណោះស្រាយចំពោះបញ្ហានោះ។

មានមួយសំនួរទៀតដែលត្រូវសួរដើម្បីរម្លឹកឡើងវិញនូវចំណេះដឹងរបស់អ្នក។ តើអ្នកមាន broadcast domains ប៉ុន្មាននៅទីនេះ?

តើ broadcast domains ជាអ្វី? យើងមិនបាននិយាយពីវានោះទេពីមុនមក។ប៉ុន្តែយើងអាចឆ្លើយសំណួរនេះបាន។ បើកុំព្យូទ័រមួយពី sales switch ចង់បញ្ជូននូវ broadcast frame មួយនោះយើងបានដឹងថាគ្រប់ Switchs ទាំងអស់និងបញ្ជូនបន្តវា។

៨-៤-របៀប configure VLANs នៅលើ Cisco Catalyst Switch

នៅក្នុងមេរៀននេះយើងនិងបង្ហាញពីរបៀប Configure VLANs នៅលើ Cisco Catalyst Switchs និងពីរបៀបកំនត់ Interfaces ទៅឲ្យ LANs មួយចំនួន។

សូមចាប់ផ្តើមជាមួយ ណេតវើកtopology ដ៏សាមញ្ញាមួយ:

កុំព្យូទ័រ A និង B ត្រូវបានភ្ជាប់ជាមួយ Switch A។ ជាដំបូងយើងត្រូវពិនិត្យមើលពី default VLAN configuration នៅលើ SwitchA:

SwitchA#show vlan

VLAN Name Status Ports

---- -------------------------------- --------- -------------------------------

1 default active Fa0/1, Fa0/2, Fa0/3, Fa0/4

Fa0/5, Fa0/6, Fa0/7, Fa0/8

Fa0/9, Fa0/10, Fa0/12

Fa0/13, Fa0/14, Fa0/22

Fa0/23, Fa0/24, Gi0/1, Gi0/2

1002 fddi-default act/unsup

1003 token-ring-default act/unsup

1004 fddinet-default act/unsup

1005 trnet-default act/unsup

គឺ VLAN 1 ជា default LAN ហើយអ្នកបានដឹងថាគ្រប់ Interfaces ទាំងអស់ត្រូវបានកំនត់មកជា VLAN 1។ ព័ត៌មានអំពី VLAN មិនត្រូវបាន Save នៅក្នុង running-config ឬ startup-config នោះទេ។ ប៉ុន្តែវាស្ថិតនៅក្នុង file ដាច់ដោយឡែកមួយហៅថា vlan.dat នៅលើ flash memory។ បើអ្នកចង់លុបព័ត៌មានអំពី VLAN នោះអ្នកអាចលុប file នេះដោយប្រើបញ្ជា delete flash:vlan.dat។ គេបាន configure នូវ IP address មួយនៅលើកុំព្យូទ័រ A និង B ដូច្នេះវាស្ថិតនៅក្នុង Subnet ដូចគ្នា។

សូមពិនិត្យមើលកុំព្យូទ័រ A និង B អាចទាក់ទងគ្នាបាន។

C:\Documents and Settings\កុំព្យូទ័រA>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms

ជា default switch configuration កុំព្យូទ័រA គឺអាចទាក់ទងជាមួយកុំព្យូទ័រ B បាន។ បើអ្នកបង្កើតនូវ VLAN ថ្មីមួយសម្រាប់កុំព្យូទ័រ A និង B នោះអ្នកត្រូវប្រើបញ្ជា:

SwitchA(config)#vlan 50

SwitchA(config-vlan)#name កុំព្យូទ័រs

SwitchA(config-vlan)#exit

នេះបង្ហាញឲ្យដឹងថាអ្នកបង្កើត VLAN ថ្មីមួយ។ បើអ្នកចង់ដាក់ឈ្មោះឲ្យវា នោះអ្នកអាចដាក់ឈ្មោះថា "computers"។

SwitchA#show vlan

VLAN Name Status Ports

---- -------------------------------- --------- -------------------------------

1 default active Fa0/1, Fa0/2, Fa0/3, Fa0/4

Fa0/5, Fa0/6, Fa0/7, Fa0/8

Fa0/9, Fa0/10, Fa0/11, Fa0/12

Fa0/13, Fa0/14, Fa0/15,

Fa0/23, Fa0/24, Gi0/1, Gi0/2

50 កុំព្យូទ័រs active

VLAN 50 ត្រូវបានបង្កើតនៅលើ SwitchA ហើយអ្នកអាចឃើញថាវាកំពុងមានសកម្មភាព។ ទោះបី ជាយ៉ាងណាក៏ដោយគ្មាន Port ណាមួយកំពុងដំណើរការនៅក្នុង VLAN 50 នោះទេ។

SwitchA(config)interface fa0/1

SwitchA(config-if)#switchport mode access

SwitchA(config-if)#switchport access vlan 50

SwitchA(config)interface fa0/2

SwitchA(config-if)#switchport mode access

SwitchA(config-if)#switchport access vlan 50

ជាដំបូងយើងត្រូវ Configure ទៅលើ Switchport នៅក្នុង Access mode ជាមួយបញ្ជា switchport mode access command។ ដោយប្រើបញ្ជា switchport access vlan យើងអាចបំលាស់ទីនៃ interfaces ទៅកាន់ VLAN មួយទៀតបាន។

SwitchA#show vlan

VLAN Name Status Ports

---- -------------------------------- --------- -------------------------------

1 default active Fa0/3, Fa0/4

Fa0/5, Fa0/6, Fa0/7, Fa0/8

Fa0/9, Fa0/10,, Fa0/12

Fa0/13, Fa0/14, Fa0/15,

Fa0/23, Fa0/24, Gi0/2

50 computers active Fa0/1, Fa0/2

ជាការល្អណាស់ដែលកុំព្យូទ័រទាំងពីរស្ថិតនៅក្នុង VLAN 50 ជាមួយគ្នា ។ សូមពិនិត្យមើលទៅលើការ Configure របស់វាដោយប្រើបញ្ជា ping ។

៨-៤-១-ពន្យល់ពី 802.1Q Encapsulation

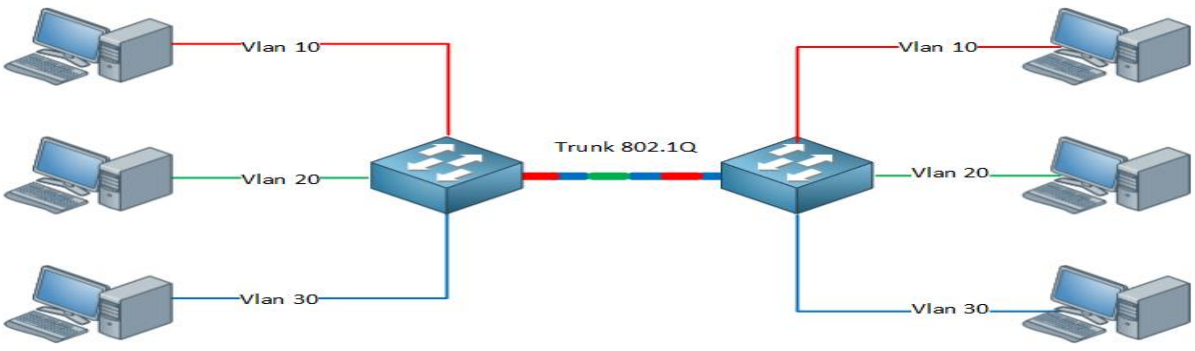
នៅពេលដែលអ្នកចង់ឲ្យចរាចរណ៍របស់ VLAN រវាង Switches ពីរដោយគ្មានបញ្ហាកើតមានឡើង ។ សូមពិនិត្យមើលរូបភាពខាងក្រោម:

នេះគឺជា Ethernet frame ធម្មតាមួយ ។ តើអ្នកឃើញផ្នែកមួយដែលអ្នកអាចកំនត់ឲ្យ VLAN នៅក្នុង Ethernet frame របស់យើងស្ថិតនៅ ?

គ្មាននោះទេ ។ ដូច្នេះតើ Switch មួយដឹងថា VLAN មួយចង់បញ្ជូនអ្វីមួយទៅឲ្យនៅពេលវាទទួល frame តាមរបៀបណា ? គ្មានមធ្យោបាយនោះទេ ។ ដូច្នេះហើយបានជាយើងត្រូវការនូវ Protocol មួយផ្សេងទៀតជួយយើង ។

បើអ្នកចង់ឲ្យចរាចរណ៍របស់ VLAN ជាមួយ Switches យើងត្រូវការនូវ trunk មួយ ។ Trunk មួយគឺជាការភ្ជាប់ធម្មតា ប៉ុន្តែវាអាចអនុញ្ញាតឲ្យចរាចរណ៍ពី VLANs ផ្សេងគ្នាអាចឆ្លងកាត់បានហើយមានវិធីសាស្ត្រក្នុងការញែកចរាចរណ៍រវាង VLANs ទាំងនោះបាន ។

ឧទាហរណ៍

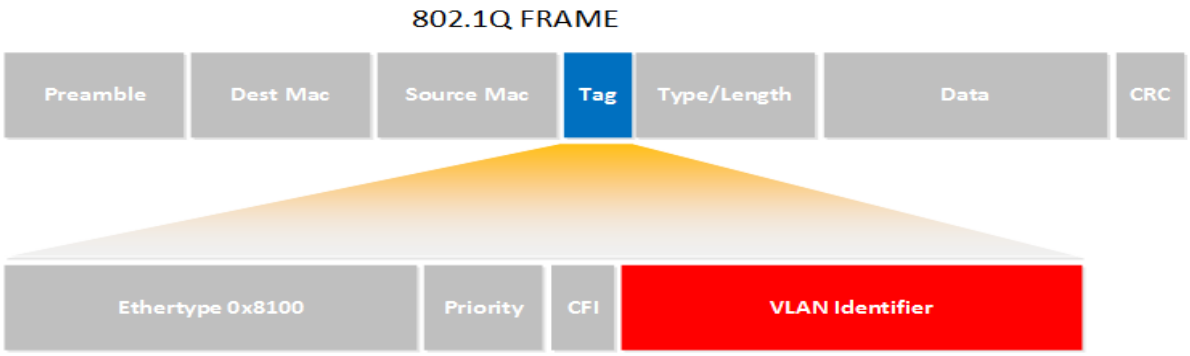


ដូចអ្នកបានឃើញស្រាប់ យើងមានកុំព្យូទ័រនៅសងខាងហើយវាស្ថិតនៅក្នុង VLANs ផ្សេងគ្នា។ ដោយការប្រើ trunks យើងអាចធ្វើឲ្យចរាចរណ៍ VLANs ទាំងអស់អាចបញ្ជូនទិន្នន័យរវាង Switches បាន។ ដោយសារតែ Ethernet frame ធម្មតាគ្មានអ្វីត្រូវបង្ហាញឲ្យដឹងថា VLAN ដែលវាស្ថិតនៅ។ មាន Trunking protocol ពីរគឺ:

802.1Q: គឺជា trunking protocol ដែលគេនិយមប្រើវា។ វាជាស្តង់ដាមួយដែលគ្រប់ផលិតផលទាំងអស់ប្រើវាបាន។

ISL: គឺជា Cisco trunking protocol ។ គ្រប់ Switches ទាំងអស់មិនប្រើវាទេ។

សូមពិនិត្យមើល 802.1Q:



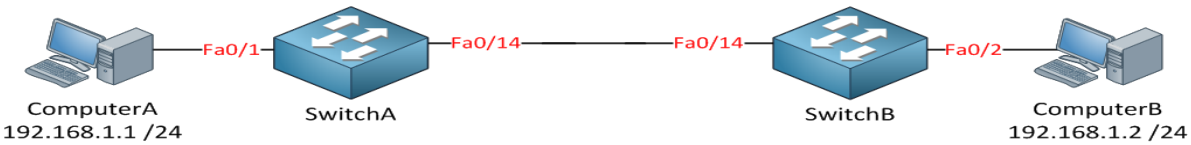
នេះគឺជាឧទាហរណ៍នៃ 802.1Q Ethernet frame។ ដូចអ្នកបានឃើញស្រាប់ វាដូចទៅនិង Ethernet frame ធម្មតាដែរ។ ប៉ុន្តែវាបានបន្ថែមនូវ tag មួយនៅកណ្តាល(មានពណ៌ខៀវ)។ នៅក្នុង tag អ្នកនឹងឃើញ “VLAN identifier” ដែលជា VLAN ដែល Ethernet frame ស្ថិតនៅជាមួយ។ នេះបង្ហាញពីរបៀបដែល Switches ដឹងថា VLAN មួយណាដែលចរាចរណ៍ជារបស់វា។

វាក៏មាន field មួយមានឈ្មោះថា “Priority” ដែលជាប្រើសម្រាប់កំនត់ពីការកំនត់អាទិភាពផ្សេងគ្នាចំពោះប្រភេទនៃចរាចរណ៍ផ្សេងគ្នា។ វាមានសារៈសំខាន់ណាស់នៅពេលដែលអ្នកមាន VLAN មួយសម្រាប់ Voice Over IP ចរាចរណ៍ និង VLAN ផ្សេងទៀតសម្រាប់ចរាចរណ៍ទិន្នន័យ។ អ្នកប្រហែលចង់ឲ្យចរាចរណ៍របស់ VoIP មានអាទិភាព។

៨-៤-២-របៀប configure trunk នៅលើ Cisco Catalyst Switch

គេត្រូវការ Trunks ដើម្បីនាំយកចរាចរណ៍របស់ VLAN ពី Switch មួយទៅកាន់ Switch មួយទៀត។ នៅក្នុងមេរៀននេះ យើងនឹងបង្ហាញពីរបៀបនៃការ Configure ទៅលើ trunk របស់ Cisco Catalyst switches។

សូមពិនិត្យទៅលើ topology ដែលអ្នកនឹងប្រើវា:



ដូចរូបខាងលើ Topolgy មានកុំព្យូទ័រមួយភ្ជាប់ជាមួយ Switch នីមួយៗ។ យើងនិងដាក់កុំព្យូទ័រនៅក្នុង VLAN តែមួយហើយបង្កើតនូវ trunk មួយរវាង Switches ពីរ។ សូមចាប់ផ្តើមជាមួយការបង្កើត VLAN មួយ:

SwitchA(config)#vlan 50

SwitchA(config-vlan)#name computers

SwitchA(config-vlan)#exit

SwitchB(config)#vlan 50

SwitchB(config-vlan)#name computers

SwitchB(config-vlan)#exit

សូមដាក់ interfaces ភ្ជាប់ជាមួយកុំព្យូទ័រនៅក្នុង VLAN ត្រឹមត្រូវ។

SwitchA(config)#interface fa0/1

SwitchA(config-if)#switchport access vlan 50

SwitchB(config)#interface fa0/2

SwitchB(config-if)#switchport access vlan 50

ជំហានបន្ទាប់គឺត្រូវបង្កើតនូវ trunk រវាង Switches ពីរ។ តាមធម្មតា Interfaces រវាង Switches ពីរអាចជា access mode ពីព្រោះថាមាន VLAN តែមួយគត់។

SwitchA(config)#interface fa0/14

SwitchA(config-if) #switchport mode trunk

Command rejected: An interface whose trunk encapsulation is "Auto" can not be configured to "trunk" mode.

SwitchB(config)#interface fa0/14

SwitchB(config-if)#switchport mode trunk

Command rejected: An interface whose trunk encapsulation is "Auto" can not be configured to "trunk" mode.

យើងបានព្យាយាមប្តូរ Interface មកជា trunk mode ជាមួយបញ្ជា switchport mode trunk ។ អាស្រ័យទៅលើ switch model អ្នកអាចឃើញមាន Error កើតមានឡើងដដែល។ បើអ្នកចង់ប្តូរ Interface មកជា trunk mode យើងក៏ត្រូវការប្តូរប្រភេទនៃ trunk encapsulation ។

SwitchA(config-if) #switchport trunk encapsulation ?

dot1q Interface uses only 802.1q trunking encapsulation when trunking

isl Interface uses only ISL trunking encapsulation when trunking

negotiate Device will negotiate trunking encapsulation with peer on interface

នេះគឺជាកន្លែងដែលអ្នកអាចជ្រើសរើសរវាង 802.1Q ឬ ISL encapsulation។ តាមលំនាំដើមរបស់ Switch វានិង ចរចារអំពីប្រភេទនៃ trunk encapsulation ។

SwitchA(config-if)#switchport trunk encapsulation dot1q

SwitchB(config-if)#switchport trunk encapsulation dot1q

ចំណាំ: បញ្ជាខាងលើនេះមិនត្រូវការទៀតទេដោយសារតែ វាមាន Protocol សម្រាប់ Encapsulation trunk តែមួយគត់គឺ 802.1Q ។

គេប្រើ

SwitchA(config-if)#switchport trunk native vlan 99

SwitchB(config-if)#switchport trunk native vlan 99

សូមប្តូរវាមកជា 802.1Q ដោយប្រើបញ្ជា switchport trunk encapsulation

SwitchA#show interfaces fa0/14 switchport

Name: Fa0/14

Switchport: Enabled

Administrative Mode: dynamic auto

Operational Mode: static access

Administrative Trunking Encapsulation: dot1q

SwitchB#show interfaces fa0/14 switchport

Name: Fa0/14

Switchport: Enabled

Administrative Mode: dynamic auto

Operational Mode: static access

Administrative Trunking Encapsulation: dot1q

ដូចអ្នកបានឃើញស្រាប់ហើយ trunk encapsulation គឺ 802.1Q ។

SwitchA(config)#interface fa0/14

SwitchA(config-if)#switchport mode trunk

SwitchB(config)#interface fa0/14

SwitchB(config-if)#switchport mode trunk

ឥឡូវនេះអ្នកអាចប្តូរមកជា switchport mode មកជា trunk ទទួលបានជោគជ័យ ។

SwitchA#show interfaces fa0/14 switchport

Name: Fa0/14

Switchport: Enabled Administrative Mode: trunk Operational Mode: trunk

Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q

SwitchB#show interfaces fa0/14 switchport

Name: Fa0/14

Switchport: Enabled Administrative Mode: trunk Operational Mode: trunk

Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q

យើងអាចបញ្ជាក់ថា យើងទទួលបាននូវ trunk មួយពីព្រោះថាវាដំណើរការនៅក្នុង “dot1q” ។

សូមព្យាយាម បើកុំព្យូទ័រ A និង B អាចទាក់ទងគ្នាបាន:

C:\Documents and Settings\កុំព្យូទ័រA>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 0ms, Maximum = 0ms, Average = 0ms

ល្អណាស់។ កុំព្យូទ័រ A និង B អាចទាក់ទងគ្នាបាន។

យើងត្រូវបង្ហាញបន្ថែមទៀត

SwitchB#show vlan

VLAN Name Status Ports

---- -------------------------------- --------- -------------------------------

1 default active Fa0/1, Fa0/3, Fa0/4, Fa0/5

Fa0/6, Fa0/7, Fa0/8, Fa0/9

Fa0/10, Fa0/11, Fa0/12, Fa0/13

Fa0/15, Fa0/22, Fa0/23, Fa0/24

Gi0/1, Gi0/2

50 computers active Fa0/2

បើអ្នកប្រើបញ្ជា show vlan ហើយមិនឃើញ Fa0/14 interface។ វាជាធម្មតាពីព្រោះថាបញ្ជា show vlan អាចបង្ហាញបានតែក្នុង access mode ប៉ុណ្ណោះមិនមែន trunk interfaces នោះទេ។

SwitchB#show interface fa0/14 trunk

Port Mode Encapsulation Status Native vlan

Fa0/14 on 802.1q trunking 1

Port Vlans allowed on trunk

Fa0/14 1-4094

Port Vlans allowed and active in management domain

Fa0/14 1,50

Port Vlans in spanning tree forwarding state and not pruned

Fa0/14 50

បញ្ជា show interface trunk មានសារៈសំខាន់ណាស់។ អ្នកអាចមើលឃើញថា interface ស្ថិតនៅក្នុង trunk mode ដែល trunk encapsulation protocol គឺប្រើ 802.1Q ឬ ISL ហើយប្រាប់ឲ្យដឹងថាវាជា native VLAN ។ យើងអាចឃើញថា VLAN 1 – 4094 ក៏ត្រូវបានកំនត់ជាមួយ trunk នេះដែរ។

យើងក៏អាចឃើញថាមានតែ VLAN 1 (native VLAN) និង VLAN 50 កំពុងមានសកម្មភាព។ អ្នកក៏អាចឃើញ VLANs មួយណាកំពុងស្ថិតនៅក្នុង forwarding state សម្រាប់ spanning-tree។

មុនពេលយើងបន្តជាមួយការ configuration នៃ VTP យើងចង់បង្ហាញឲ្យឃើញថាមាន access និង trunk interfaces:

SwitchB#show interface fa0/2 switchport

Name: Fa0/2

Switchport: Enabled

Administrative Mode: static access

Operational Mode: static access

Interface មួយអាចស្ថិតនៅក្នុង Access mode ឬ trunk mode។ interface ខាងលើត្រូវបានភ្ជាប់ជាមួយកុំព្យូទ័រ B ហើយអ្នកអាចឃើញថា operational mode គឺ “static access” មានន័យថាវាស្ថិតនៅក្នុង access mode។

SwitchB#show interfaces fa0/14 switchport

Name: Fa0/14

Switchport: Enabled

Administrative Mode: trunk

Operational Mode: trunk

នេះគឺជា trunk interface ត្រូវបានភ្ជាប់ជាមួយ SwitchA ។ អ្នកក៏អាចឃើញ operational mode គឺជា trunk mode។

SwitchB(config-if)#switchport mode ?

access Set trunking mode to ACCESS unconditionally

dot1q-tunnel set trunking mode to TUNNEL unconditionally

dynamic Set trunking mode to dynamically negotiate access or trunk

private-vlan Set private-vlan mode

trunk Set trunking mode to TRUNK unconditionally

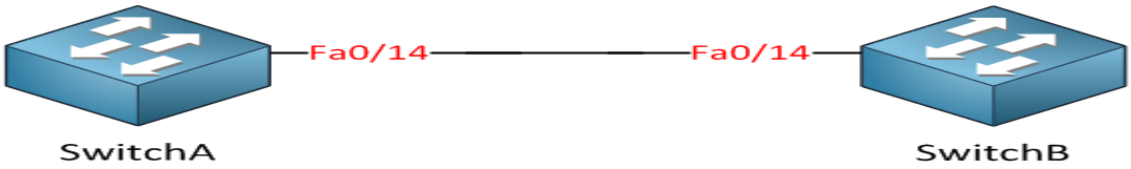
បើយើងចូលទៅកាន់ interface configuration ដើម្បីប្តូរ switchport mode អ្នកក៏អាចឃើញថាវាមាន Options ច្រើនជាង access ឬ trunk mode។ វាក៏មាន dynamic មួយផងដែរ។

SwitchB(config-if)#switchport mode dynamic ?

auto Set trunking mode dynamic negotiation parameter to AUTO

desirable Set trunking mode dynamic negotiation parameter to DESIRABLE

យើងអាចជ្រើសរើសយកវាង dynamic auto និង dynamic desirable។ Switch របស់យើងរកឃើញវាជាស្ងួយប្រវត្តិបើ Interface បានក្លាយជា access ឬ trunk port។ តើខុសគ្នារវាង dynamic auto និង dynamic ?

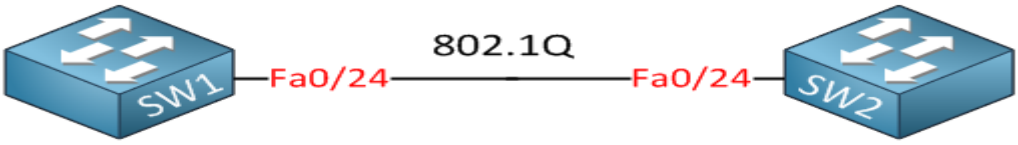


៨-៤-៣-802.1Q Native VLAN នៅលើ Cisco IOS Switch

IEEE 802.1Q trunking protocol បានពិពណ៌នាអំពី “native VLAN”។ គ្រប់ចរាចរណ៍ទាំងអស់ដែលបានបញ្ជូននិងទទួលនៅលើ interface ត្រូវបាន Configure សម្រាប់ 802.1Q ដោយមិនបាន tag នៅលើ Ethernet frame។ នៅពេលអ្នកពិនិត្យមើលនៅក្នុង Wireshark អ្នកនឹងឃើញថាវាដូចទៅនិង Ethernet frame ធម្មតាដែរ។

នៅពេលដែល Cisco Switches ទទួលនូវ Ethernet frame ដោយគ្មាន tag នៅលើ 802.1Q enabled interface វានិងសន្មត់ថាវាជារបស់ native VLAN។ ចំពោះហេតុផលនេះ អ្នកត្រូវប្រាកដថា native VLAN ដូចគ្នានៅទាំងសងខាង។ តាមលំនាំដើមនៃ native VLAN គឺជា VLAN 1។ ប៉ុន្តែយើអាចប្តូរវាបានបើអ្នកចង់ប្តូរវា។

សូមពិនិត្យមើលឧទាហរណ៍ យើងមាន Switches ពីរ។



យើងនិង configure ទៅលើ 802.1Q trunk រវាង Switches ទាំងពីរនោះ ។ ដូច្នេះអ្នកអាចឃើញ native VLAN:

SW1(config)#interface Fastethernet 0/24

SW1(config-if)#switchport trunk encapsulation dot1q

SW1(config-if)#switchport mode trunk

SW2(config)#interface Fastethernet 0/24

SW2(config-if)#switchport trunk encapsulation dot1q

SW2(config-if)#switchport mode trunk

យើងអាចផ្ទៀងផ្ទាត់ចំពោះ trunk configuration ហើយឃើញពី native VLAN ដូចខាងក្រោម:

SW1#show interface fastEthernet 0/24 trunk

Port	Mode	Encapsulation	Status	Native vlan
Fa0/24	on	802.1q	trunking	1

Port	Vlans allowed on trunk
Fa0/24	1-4094

Port	Vlans allowed and active in management domain
Fa0/24	1,10,12-13,20,23,34,100,123

Port	Vlans in spanning tree forwarding state and not pruned
Fa0/24	1,10,12-13,20,23,34,100,123

SW2#show interfaces fastEthernet 0/24 trunk

Port	Mode	Encapsulation	Status	Native vlan
Fa0/24	on	802.1q	trunking	1

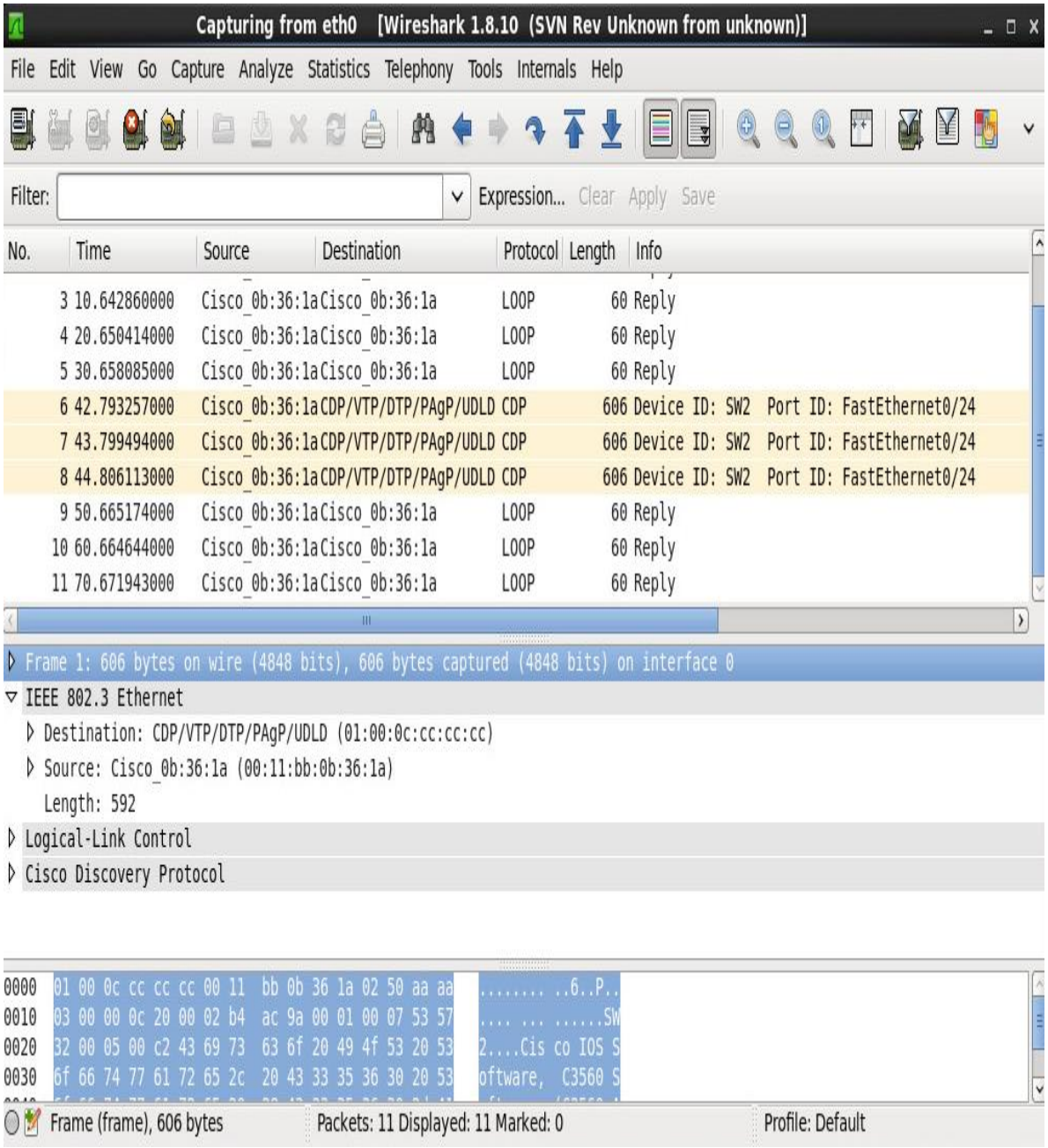
Port	Vlans allowed on trunk
Fa0/24	1-4094

Port	Vlans allowed and active in management domain
Fa0/24	1,10,12-13,20,23-24,30

Port	Vlans in spanning tree forwarding state and not pruned
Fa0/24	1,10,12-13,20,23-24,30

ដូចអ្នកបានឃើញខាងលើស្រាប់ថា trunk កំពុងដំណើរការ។ យើងប្រើ 802.1Q encapsulation និង native VLAN គឺ 1។ តើចរាចរណ៍ប្រភេទណាកំពុងដំណើរការនៅលើ native VLAN?

សូមពិនិត្យមើលទៅលើ wireshark ដែលបានចាប់យក trunk!

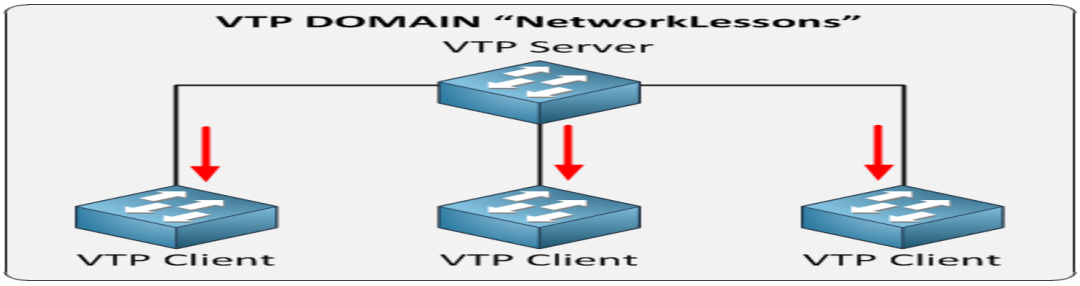


ដូចអ្នកបានឃើញពីការគ្រប់គ្រងនៃ Protocol ដូចជា CDP (Cisco Discovery Protocol) ត្រូវបានបញ្ជូននៅលើ native VLAN។ សម្រាប់សុវត្ថិភាព វាជាគំនិតដ៏ល្អដើម្បីប្តូរ native VLAN ពី VLAN 1 មកជាឈ្មោះផ្សេង។

៨-៥-សេចក្តីផ្តើមចំពោះ VTP (VLAN Trunking Protocol)

ឧបមាថាអ្នកមាន ណេតវើកមាន 20 switches និង 50 VLANs។ តាមធម្មតាអ្នកត្រូវតែ Configure Switch នីមួយៗដាច់ដោយឡែកពីគ្នាហើយបង្កើត VLANs នៅលើ Switch នីមួយៗ។ កិច្ចការនេះគេហៅថា VTP (VLAN Trunking Protocol)។

VTP និងអនុញ្ញាតឲ្យអ្នកបង្កើតនូវ VLANs នៅលើ Switch ហើយគ្រប់ Switches ផ្សេងទៀតត្រូវតែនិងធ្វើឲ្យដូចគ្នា (synchronize) ដោយខ្លួនវា។



យើងមាន VTP server មួយដែលជា Switch អ្នកត្រូវបង្កើត កែប្រែឬលុប VLANs ។ ចំពោះ Switches ផ្សេងទៀតគឺជា VTP clients។ ការ Configure VTP មានលេខមួយដែលនិងកើនឡើងនៅពេលអ្នកប្តូរវា។ គ្រប់ពេលដែលអ្នកធ្វើការផ្លាស់ប្តូរនៅលើ VTP server វានិងធ្វើឲ្យ VTP clients ទាំងអស់ដូចគ្នា។ អ្នកក៏អាចមាន VTP servers ជាច្រើនផងដែរ ពីព្រោះថាវាដើរតួនាទីជា VTP client ដែរ។ អ្នកអាចធ្វើការផ្លាស់ប្តូរនៅលើ Switches ជាច្រើននៅក្នុង ណេតវើកមួយ។ ដើម្បីធ្វើឲ្យ VTP ដំណើរការបាន អ្នកត្រូវតែ setup នូវ VTP domain មួយ។

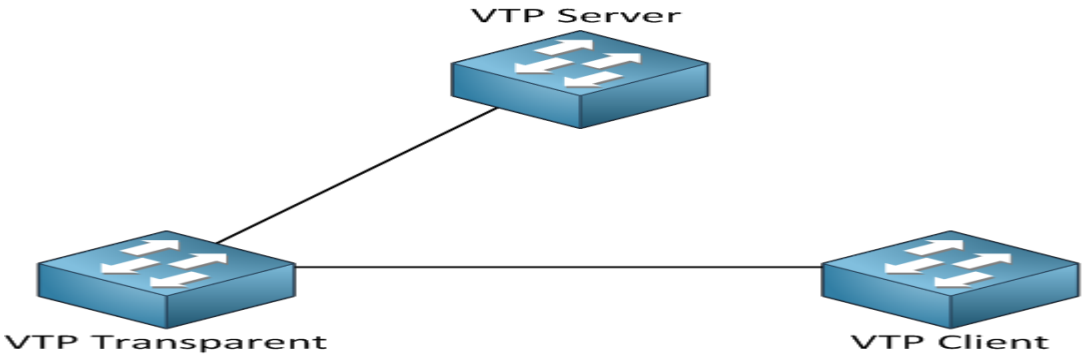
នេះគឺជាអ្វីដែលយើងទើបតែបានពិពណ៌នាអំពីវា:

១-បង្កើត VTP / កែប្រែ / លុប VLANs

២-រាល់ពេលនៃការកែប្រែលេខនិងមានការកើនឡើង

៣-VTP clients និងធ្វើឲ្យដូចគ្នាដោយខ្លួនវាជាមួយព័ត៌មានចុងក្រោយបំផុត

ក្រៅពី VTP server និង VTP clients ក៏មាន VTP transparent មួយដែលជាការពិបាកមួយដែរ។ សូមពិនិត្យមើលពីឧទាហរណ៍



VTP transparent និងបញ្ជូនបន្តចំពោះការផ្សព្វផ្សាយ។ ប៉ុន្តែមិនបានធ្វើឲ្យដូចគ្នានោះទេ។ អ្នកអាចបង្កើត VLANs ជា local បាន។ ទោះបីជាមោនអាចធ្វើទៅបាននៅលើ VTP client ។ ឧបមាថា អ្នកបង្កើត VLAN 20 នៅលើ VTP server នោះវានិងមានកើតឡើង:

១-អ្នកបង្កើត VLAN 20 នៅលើ VTP server

២-លេខរបស់វា(revision) និងកើនឡើង

៣- VTP server និងបញ្ជូនបន្តនូវការផ្សព្វផ្សាយព័ត៌មានចុងក្រោយដែលនិងទៅដល់ VTP transparent switch.

៤-VTP transparent និងមិនធ្វើឲ្យដូចគ្នានោះទេ។ ប៉ុន្តែនិងបញ្ជូនបន្តចំពោះការផ្សព្វផ្សាយមកកាន់ VTP client

៥- VTP client និងធ្វើឲ្យដូចគ្នាជាមួយព័ត៌មានចុងក្រោយ

នេះគឺជាការបង្ហាញពី 3 VTP modes:

	VTP Server	VTP Client	VTP Transparent
ate/Modify/Delete VLANs	Yes	No	Only local
chronizes itself	Yes	Yes	No
wards advertisements	Yes	Yes	Yes

តើយើងគួរប្រើ VTP ឬទេ ? វាហាក់បីដូចជាល្អ។ ប៉ុន្តែ VTP មានហានិភ័យចំពោះសុវត្ថិភាពខ្ពស់បន្តិច។ បញ្ហាគឺ VTP អាចជា VTP server ហើយក៏អាចជា VTP client ផងដែរហើយ VTP client និងធ្វើឲ្យដូចគ្នាជាមួយលេខ revision ដែលមានកម្រិតខ្ពស់បំផុត។

ស្ថានភាពខាងក្រោមនេះនិងកើតមានឡើងចំពោះ VTP:

អ្នកមាន ណេតវើកមួយដែលមាន VTP server តែមួយគត់ហើយមាន VTP client Switches ចំនួនពីរ។ អ្វីៗដំណើរការបានល្អ ប៉ុន្តែថ្ងៃមួយអ្នកចង់សាកលក្បងពីកិច្ចការមួយចំនួនហើយសម្រេចចិត្តយក VTP client មួយ ចេញពី ណេតវើកហើយដាក់វាចូលទៅក្នុង Lab ។ នៅពេលអ្នកយក VTP client Switch ចេញពី ណេតវើកអ្នកត្រូវ:

១-អ្នកត្រូវ Configure វា ។ ដូច្នេះវាមិនមែនជា VTP client ទៀតទេ វាគឺជា VTP server មួយ

២-អ្នកសាកល្បងជាមួយ VTP ដោយបង្កើត VLANs ខ្លះ ហើយកែប្រែខ្លះ

៣-គ្រប់ពេលដែលអ្នកបង្កើតការផ្លាស់ប្តូរ នោះ លេខ revision ត្រូវបានកើនឡើង

៤- អ្នកបានសាកល្បង ហើយលុប VLANs ទាំងអស់ចោល

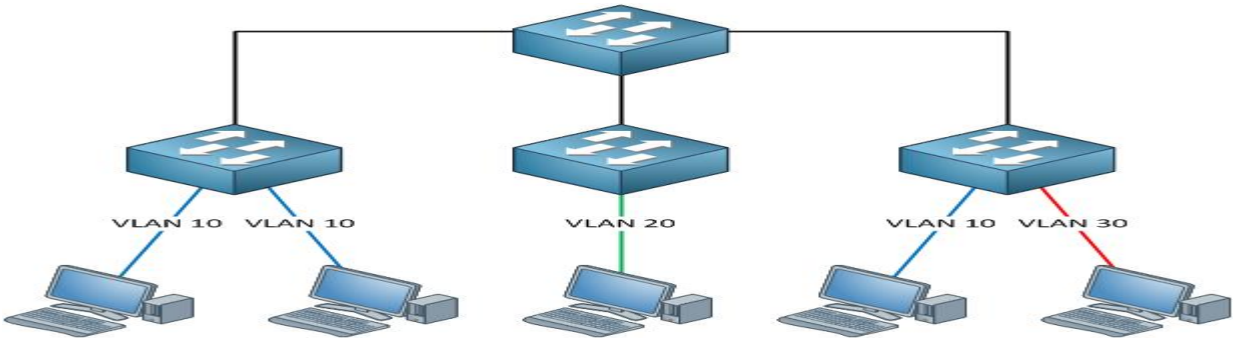
៥-អ្នក Configure Switch ពី VTP server មកជា VTP client

៦-អ្នកភ្ជាប់មកកាន់ Switch នៅក្នុងផលិតកម្ម ណេតវើក

តើអ្នកគិតថាលទ្ធផលយ៉ាងណា ? លេខ revision នៃ VTP នៅលើ Switch ដែលយើងបានសាកល្បងជាមួយវាគឺមានការកើនឡើងកម្រិតខ្ពស់នៅលើ Switches នៃផលិតកម្ម Network។ VTP client និងផ្សព្វផ្សាយព័ត៌មានរបស់វាទៅឲ្យ Switches ផ្សេងទៀត។ វាបានធ្វើឲ្យព័ត៌មានចុងក្រោយដូចគ្នាហើយបញ្ជូនទៅឲ្យ VLANs

ទាំងអស់។ VTP client មួយអាច Overwrite ទៅលើ VTP server មួយបើលេខ revision កើនឡើងខ្ពស់ជាង ដោយសារតែ VTP server មួយក៏អាចជា VTP client បានដែរ។

អ្នកបានដឹងហើយវាពិបាក ប៉ុន្តែនេះជាវិធីដែលវាដំណើរការហើយគ្រោះថ្នាក់ពីព្រោះថាអ្នកនិងបាត់បង់ព័ត៌មាន អំពី VLAN ។ Interface របស់អ្នកមិនអាចក្លាយជា VLAN1 បានទេ។ ចំណុចមួយទៀតនៃ VTP គឺមានដូចក្នុងរូបភាព។

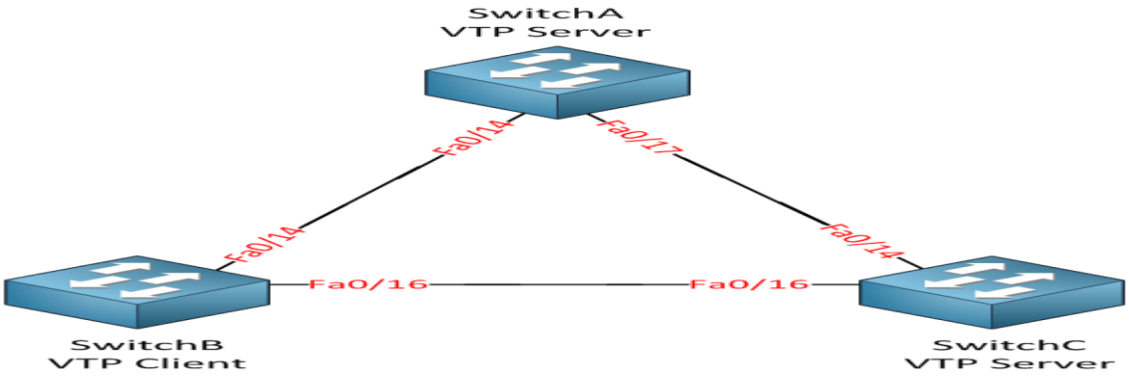


អ្នកបានឃើញថាយើងមានកុំព្យូទ័រនៅក្នុង VLAN 10, 20 និង 30 ។ ការភ្ជាប់រវាង switches គឺជា trunks ដែល កំពុងប្រើ 802.1Q protocol ហើយនាំយកគ្រប់ចរាចរណ៍របស់ VLAN ។ កុំព្យូទ័រមួយនៅក្នុង VLAN 10 បញ្ជូន broadcast frame មួយ។ តើអ្នកគិតថា broadcast frame នេះទៅទីណា ?

Broadcast frames ត្រូវតែបញ្ជូនដោយ Switches ទាំងអស់ហើយ trunks របស់យើងកំពុងទទួលនូវគ្រប់ ចរាចរណ៍របស់ VLANs ។ broadcast ទៅគ្រប់ទីកន្លែងទាំងអស់ ។ បើអ្នកពិនិត្យមើលពី Switch នៅកណ្តាលវិញ តើអ្នក បានឃើញកុំព្យូទ័រនៅក្នុង VLAN 10 ដែរឬទេ ? មានតែ VLAN 20 មួយគត់ដែល broadcast នេះត្រូវបានធ្វើឲ្យខាត bandwidth ។

ដោយបើក VTP pruning យើងនិងដឹងថាគ្មានចរាចរណ៍របស់ VLAN ដែលមិនចាំបាច់នៅលើ Trunks នោះទេនៅ ពេលដែលគ្មានកុំព្យូទ័រណាមួយនៅក្នុង VLAN ពិសេស ។ អាស្រ័យទៅលើ Switch model VTP pruning ត្រូវបាន បើកឬបិទតាមលំនាំដើម។

សូមពិនិត្យមើលចំពោះការ Configure នៃ VTP ។ គេប្រើ Switches ចំនួនបី។ យើងនិងលុប VLAN database និង startup-configuration នៅលើ switches ទាំងអស់ ។



SwitchA#show vtp status

VTP Version : running VTP1 (VTP2 capable)

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 5

VTP Operating Mode : Server

VTP Domain Name :

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x57 0xCD 0x40 0x65 0x63 0x59 0x47 0xBD

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Local updater ID is 0.0.0.0 (no valid interface found)

SwitchB#show vtp status

VTP Version : running VTP1 (VTP2 capable)

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 5

VTP Operating Mode : Server

VTP Domain Name :

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x57 0xCD 0x40 0x65 0x63 0x59 0x47 0xBD

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Local updater ID is 0.0.0.0 (no valid interface found)

SwitchC#show vtp status

VTP Version : 2

Configuration Revision : 0

Maximum VLANs supported locally : 1005

Number of existing VLANs : 5

VTP Operating Mode : Server

VTP Domain Name :

VTP Pruning Mode : Disabled

VTP V2 Mode : Disabled

VTP Traps Generation : Disabled

MD5 digest : 0x57 0xCD 0x40 0x65 0x63 0x59 0x47 0xBD

Configuration last modified by 0.0.0.0 at 0-0-00 00:00:00

Local updater ID is 0.0.0.0 (no valid interface found)

ឧទាហរណ៍នៃ VTP

ដើម្បីនាំយកចរាចរណ៍នៃ VLAN ជាដំបូងត្រូវ Configure នៅលើ Switch។ ឧបមាថាបើ User ម្នាក់ចង់បញ្ជូន Frame មួយពីប្រភពដើមកាន់គោលដៅហើយផ្លូវដែលខ្លីជាងគេមាន១000 switches ។ ដើម្បីដំណើរការចំពោះ Frame ណាមួយនៃ VLAN បានគឺគេត្រូវ Configure ទៅលើ VLAN ជាមុនសិនហើយត្រូវ Configure ទៅលើ 1000 Switches ។ កិច្ចការនេះសម្រាប់ Admin មិនអាចធ្វើបាននោះទេ។ ដូច្នេះត្រូវ ប្រើ VTP ។

VTP គឺជា Protocol របស់ Cisco ដែលត្រូវបានគេប្រើសម្រាប់រក្សានូវ consistency តាមរយៈ ណេតវើកឬ user ដែលអាចនិយាយថាធ្វើឲ្យព័ត៌មានរបស់ VLAN ដូចគ្នានៅក្នុង VTP domain ។

តម្រូវការចាំបាចរបស់ VTP រវាង Switches មានដូចខាងក្រោម:

១-VTP version ត្រូវតែដូចគ្នានៅលើ Switch ដែលអ្នកប្រើប្រាស់ចង់ Configure

២-ឈ្មោះរបស់ VTP domain ត្រូវតែដូចគ្នានៅលើ Switches

៣-ត្រូវតែមាន Switch មួយជា Server

៤-Authentication ត្រូវតែត្រូវគ្នាបើមាន

Configuration – User will first make the switch VTP server

```
Switch# config terminal

Switch(config)#vtp mode server
```

Now, User has to make a VTP domain assign a password for authentication.

```
Switch(config)#vtp domain geeksforgeeks

Switch(config)#vtp password hardwork
```

User can verify the configuration by:

```
Switch(config)#do should vtp password

Switch(config)#do show vtp
```

Configuration

តាមលំនាំដើម Switch ត្រូវបានកំនត់ជា Server។ ដូច្នេះអ្នកប្រើប្រាស់អាចប្តូរវាជា Client ដោយ:

```
Switch(config)#vtp mode client
```

Transparent

នៅក្នុង Mode នេះបញ្ជូនបន្តតែអ្វីដែលបានសង្ខេបពី VTP នៃការផ្សព្វផ្សាយតាមរយៈនៃ trunk link ។

Configuration

អ្នកប្រើប្រាស់អាចប្តូរ Mode មកជា transparent ដោយ

```
Switch(config)#vtp mode transparent
```

Configuration Revision Number

លេខនៃ Configuration revision មានប្រវែង 32 bits ដែលបង្ហាញពីកម្រិតនៃ Revision សម្រាប់ VTP Packet មួយ។ អ្នកប្រើប្រាស់អាចពិនិត្យទៅលើលេខនៃ Configuration Revision ដោយៈ

```
switch(config)#do show vtp status
```

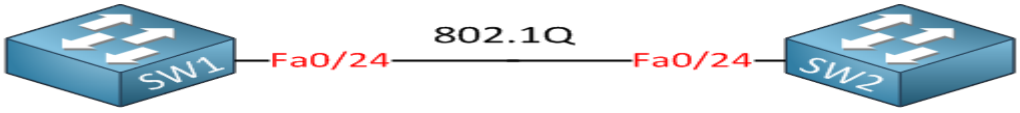
Cisco DTP (Dynamic Trunking Protocol)

ការចរចារ

នៅក្នុងមេរៀននេះ យើងពិនិត្យមើលទៅលើ DTP(Dynamic Trunking Protocol) negotiation។ DTP ត្រូវបានប្រើតាមធម្មតានៅលើ Cisco IOS switches ដើម្បីចរចារគ្នាបើ Interface បានក្លាយជា access port ឬ trunk មួយ។

តាម DTP លំនាំដើមត្រូវបានបើកហើយ Interfaces នៃ Switches និងស្ថិតនៅក្នុង “dynamic auto” ឬ “dynamic desirable” mode។ នេះមានន័យថានៅពេលណាក៏ដោយវាទទួលបាននូវ DTP packet មួយដែលស្នើបង្កើតជា Trunk មួយ Interface និងស្ថិតនៅក្នុង trunk mode។

សូមពិនិត្យមើលពីការចរចារនៃ DTP និងពីរបៀបបិទវា។ យើងនិងប្រើ Switches ពីរ:



យើងមិនបាន Configure អ្វីនៅលើ Switches នោះទេ។ សូមពិនិត្យទៅលើការកំនត់ជា Default មានដូចខាងក្រោម:

SW1#show interfaces fa0/24 switchport

Name: Fa0/24

Switchport: Enabled

Administrative Mode: dynamic auto

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

SW2#show interfaces fastEthernet 0/24 switchport

Name: Fa0/24

Switchport: Enabled

Administrative Mode: dynamic auto

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: On

ដោយគ្មានការ Configure អ្វីទាំងអស់នៅលើ Interfaces យើងកំពុងប្រើ dynamic auto mode ហើយផ្តល់លទ្ធផលឲ្យ interfaces ស្ថិតនៅក្នុង access mode។ អាស្រ័យទៅលើប្រភេទនៃ Switch និង IOS version តាមលំនាំដើមវាជា “dynamic auto” ឬ “dynamic desirable”។ Switches ដែលប្រើជា Cisco Catalyst 3560 switches។

មានវិធីពីរយ៉ាងដើម្បីបិទការចរចាររបស់ DTP:

Configure interface សម្រាប់ access mode

ប្រើបញ្ជា switchport nonegotiate នៅលើ interface

ការ Configure ទៅលើ Interface សម្រាប់ trunk មិនអាចបិទការចរចាររបស់ DTP នោះទេ។ សូមពិនិត្យឧទាហរណ៍ ។ ជាដំបូង យើងនិង Configure Interface សម្រាប់ Access mode:

SW1(config)#interface fastEthernet 0/24

SW1(config-if)#switchport mode access

SW2(config)#interface fastEthernet 0/24

SW2(config-if)#switchport mode access

នៅពេលយើងពិនិត្យមើលទៅលើការកំនត់នៃ Switchport យើងអាចឃើញថាការចរចាររបស់ DTP ត្រូវបានបិទ ឥឡូវនេះ។

SW1#show interfaces fastEthernet 0/24 switchport

Name: Fa0/24

Switchport: Enabled

Administrative Mode: static access

Operational Mode: static access

Administrative Trunking Encapsulation: negotiate

Operational Trunking Encapsulation: native

Negotiation of Trunking: Off

ដូច្នេះការ Configure ទៅលើ Interface ដើម្បីប្រើ access mode សម្រាប់បិទការចរចាររបស់ DTP ។ តើការបង្កើត trunk មួយយ៉ាងណា ?

ជំពូកទី៩

Spanning-Tree

៩-១-សេចក្តីផ្តើមចំពោះ Spanning Tree

Spanning-tree គឺជា protocol មួយដែលដំណើរការនៅលើ Switches របស់យើងហើយអាចជួយយើងក្នុងការដោះស្រាយបញ្ហារបស់ loops។ Spanning-tree គឺជា protocol មួយដែលអ្នកត្រូវតែយល់នៅពេលដែលអ្នកក្លាយជាវិស្វករ ណេតវើកម្នាក់ហើយអ្នកនិងជួបប្រទះបើអ្នកប្រឈមជាមួយការប្រលងនៃ Cisco CCNA ។

មេរៀននេះណែនាំឲ្យស្គាល់ពី spanning-tree។ អ្នកនិងសិក្សាពីវា ដឹងពីរបៀបវាដំណើរការនិងដឹងពីរបៀបពិនិត្យមើលទៅលើ spanning-tree topology នៅលើ Cisco switches។

៩-២-ហេតុអ្វីបានជាអ្នកត្រូវការនូវ spanning-tree?

តើ loop មួយជាអ្វី? ហើយអ្នកទទួលវាបានតាមរបៀបណា?

សូមពិនិត្យមើលឧទាហរណ៍:

នៅក្នុងរូបភាពខាងលើនេះ យើងមាន Switches ពីរ។ Switches ទាំងពីរត្រូវបានភ្ជាប់ជាមួយគ្នាជាមួយខ្សែ ណេតវើកតែមួយ។ ដូច្នេះមានចំណុចបរាជ័យតែមួយ។ ដើម្បីគេចផុតពីភាពបរាជ័យនេះបាន យើងត្រូវបន្ថែមខ្សែមួយទៀត។

ជាមួយខ្សែបន្ថែមមួយទៀត យើងមាន redundancy។ ចំពោះ redundancy វិញក៏មាន loops ដែរ។ ហេតុអ្វីបានជាមាន loop កើតមានឡើង?

សូមពិនិត្យមើល

១-កុំព្យូទ័រ A បញ្ជូននូវ ARP request ដើម្បីស្វែងរក MAC address របស់កុំព្យូទ័រ B ។ ARP request គឺជា broadcast frame មួយ។

២-Switch A និងបញ្ជូនបន្តនូវ broadcast frame នេះនៅលើគ្រប់ interfaces ទាំងអស់ លើកលែងតែ Interface ដែលវាទទួល frame

៣- Switch B និងទទួលនូវ broadcast frames ទាំងពីរ

តើ Switch B ធ្វើអ្វីខ្លះជាមួយ broadcast frames?

១-វានិងបញ្ជូនបន្តទៅឲ្យគ្រប់ Interfaces ទាំងអស់ លើកលែងតែ Interface ដែលវាទទួល frame

២-នេះមានន័យថា frame ដែលត្រូវបានទទួលនៅលើ interface Fa0/0 និងត្រូវបានបញ្ជូនបន្តទៅ Interface Fa1/0

៣- Frame ដែលត្រូវបានទទួលនៅលើ Interface Fa1/0 និងត្រូវបានបញ្ជូនបន្តទៅ Interface Fa0/0

តើអ្នកដឹងថាវាទៅទីណាទេ? យើងមាន loop កើតមានឡើង។ Switches ទាំងសងខាងនិងនៅតែបន្តបញ្ជូនបន្ត ហើយបន្តរហូតទាល់តែមាន loop កើតមានឡើង។

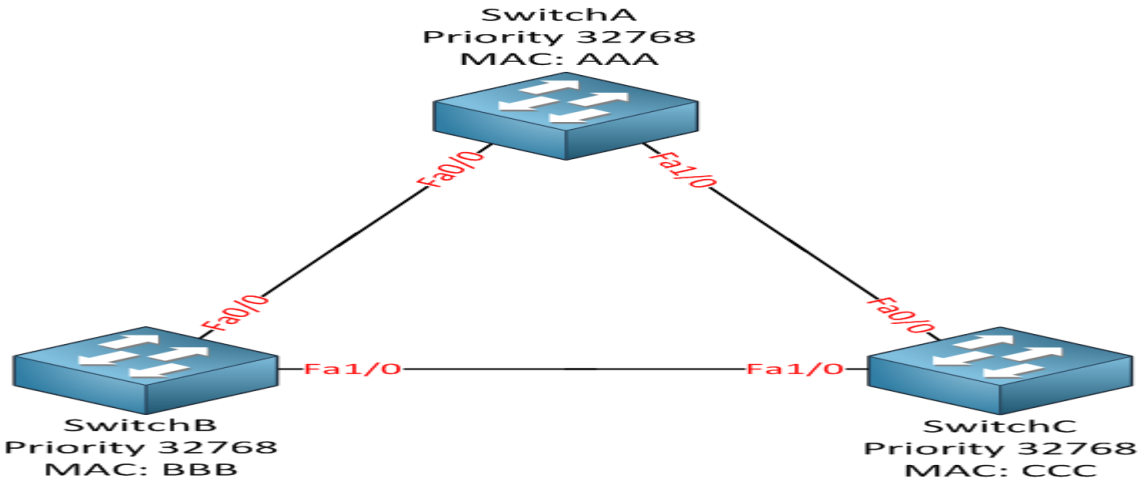
អ្នកអាចដោះស្រាយបញ្ហា loop នេះដោយផ្តាច់ខ្សែចោល។

ចំពោះ Ethernet frame វិញគ្មាន TTL (Time to Live) នោះទេ។ ដូច្នេះវានិងមាន loop កើតឡើងជារៀងរហូត។ ក្រៅពី ARP requests មាន frames ជាច្រើនដែលជា broadcast ។ ឧទាហរណ៍ នៅពេលដែល Switch មិនបានស្គាល់ពី MAC address របស់គោលដៅទេ វានិងបញ្ជូនទៅឲ្យទាំងអស់គ្នា។

៩-២-១-របៀបដោះស្រាយបញ្ហា loops ជាមួយ Spanning-tree

Spanning-tree និងជួយយើងក្នុងការបង្កើតនូវ loop-free topology មួយដោយបិទចោលនូវ interfaces ខ្លះ។

សូមពិនិត្យទៅលើឧទាហរណ៍ស្តីអំពីរបៀបដែល Spanning-tree ដំណើរការ។



យើងមាន Switches ចំនួន៣ហើយដូចអ្នកបានដឹងស្រាប់ យើងមានបន្ថែមនៃ redundancy ដោយភ្ជាប់ Switches ជារាងត្រីកោណមួយ។ នេះមានន័យថា យើងមាន loop នៅទីនេះ។ យើងបានបន្ថែម MAC addresses:

Switch A: MAC AAA

Switch B: MAC BBB

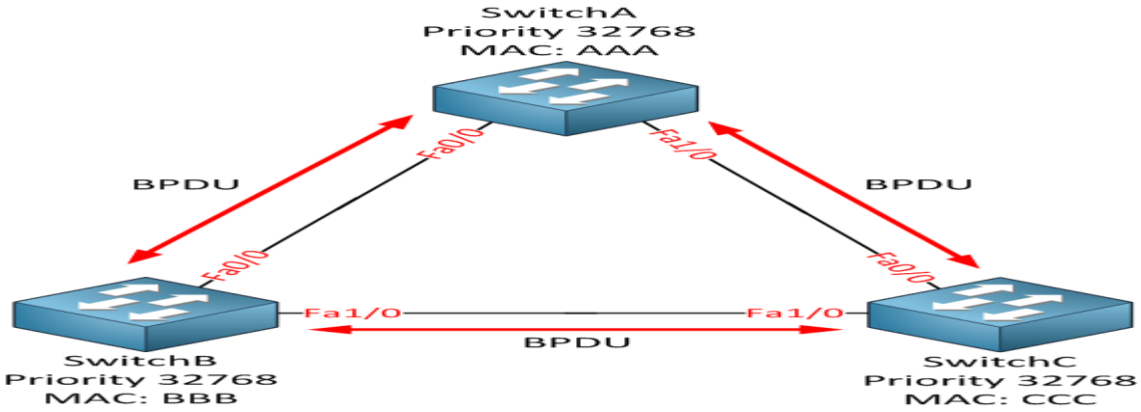
Switch C: MAC CCC

ដោយសារតែ Spanning tree ត្រូវបានបើក។ switches ទាំងអស់និងបញ្ជូននូវ frame ពិសេសមួយទៅគ្នាទៅវិញ ទៅមកហៅថា BPDU (Bridge Protocol Data Unit)។ នៅក្នុង BPDU មានព័ត៌មានពីរសំខាន់ដែល Spanning tree ត្រូវការគឺ:

MAC address

Priority

ចំពោះ MAC address និង priority បង្កើតបានជា bridge ID។ BPDU ត្រូវបានបញ្ជូនរវាង Switches ដូចបង្ហាញក្នុងរូបភាពខាងក្រោម:



Spanning-tree ត្រូវការនូវ bridge ID សម្រាប់ការគណនារបស់វា។ សូមពិនិត្យមើលពីរបៀបវាដំណើរការ:

ជាដំបូងគ្រប់ Spanning tree និងជ្រើសរើសយក root bridge មួយ។ root-bridge នេះគឺមាន “bridge ID” ប្រសើរជាងគេ។

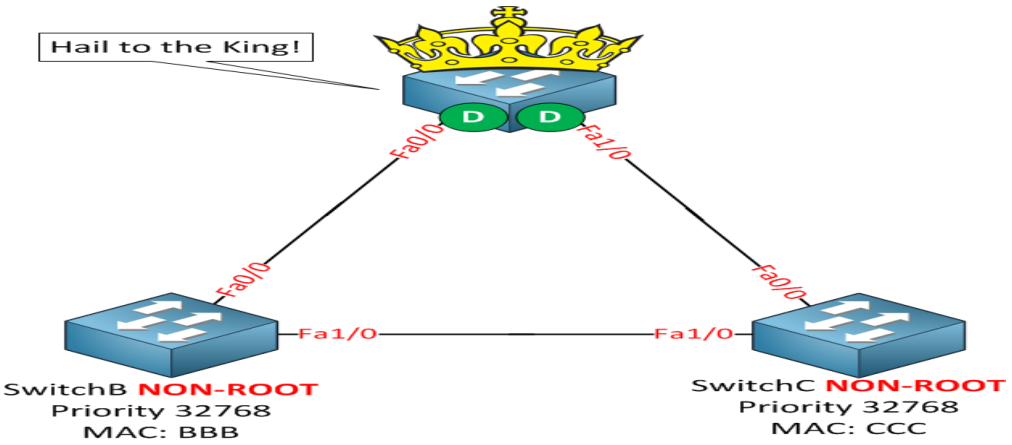
Switch ដែលមាន Bridge ID តូចជាងគេបំផុតគឺជា ID ដែលល្អប្រសើរជាងគេ

តាមលំនាំដើម Priority គឺ 32768 ប៉ុន្តែយើងអាចប្តូរវាបាន

ដូច្នេះតើមួយណាក្លាយជា root bridge?

នៅក្នុងឧទាហរណ៍របស់យើង SwitchA និងក្លាយជា root bridge។ Priority និង MAC address បង្កើតបានជា bridge ID។ ដោយសារតែ priority ដូចគ្នាទាំងអស់នៅគ្រប់ Switches ។ ដូច្នេះ MAC address គឺជាជម្រើសនៃការកំនត់។ Switch A មាន MAC address ទាប់ជាងគេបំផុត វាក្លាយជា bridge ID ល្អប្រសើរជាងគេបំផុត។ ដូចនេះវាក្លាយជា root bridge។

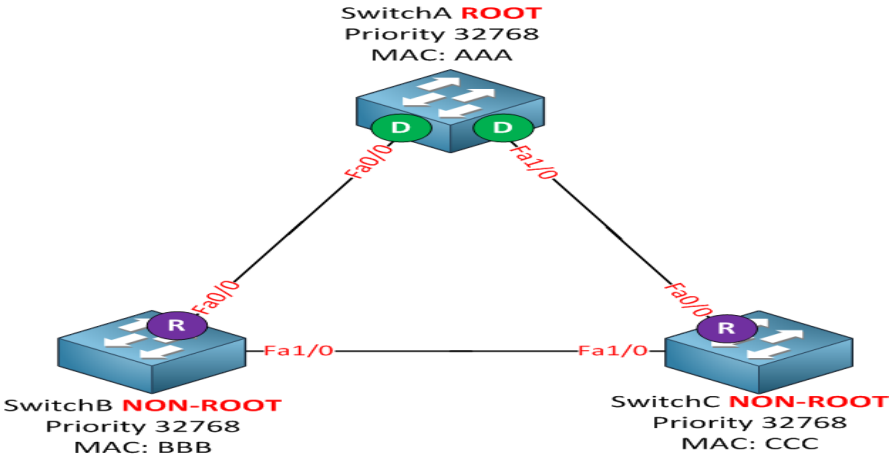
Port នៅលើ root bridge ត្រូវបានក្លាយជា designated ។ មានន័យថាវាស្ថិតនៅក្នុងលក្ខណៈ forwarding ។ សូមពិនិត្យមើលទៅលើរូបភាព:



ដូចរូបភាពខាងលើ SwitchA ត្រូវបានជ្រើសរើសជា root bridge ហើយ “D” នៅលើ Interfaces ក្លាយជា designated ។

តទៅវានេះយើងបានយល់ព្រមទៅលើ root bridge ។ ជំហានបន្ទាប់គឺ “non-root” bridges (ដូច្នេះគ្រប់ Switches ទាំងអស់មិនមែនជា root) ដែលត្រូវរកផ្លូវដែលខ្លីបំផុតមកកាន់ root bridge ។

ផ្លូវដែលខ្លីបំផុតមកកាន់ root bridge ហៅថា “root port” ។ សូមពិនិត្យមើលឧទាហរណ៍:



យើងបានដាក់សញ្ញា “R” សម្រាប់ “root port” នៅលើ Switch B និង Switch C ដែល Fa0/0 interface គឺជាផ្លូវដែលខ្លីជាងគេបំផុតមកកាន់ root bridge ។ នៅក្នុងឧទាហរណ៍ “ផ្លូវដែលខ្លីជាងគេបំផុត” នៅក្នុង Spanning-tree គឺមានន័យថាវានិងពិនិត្យទៅលើល្បឿននៃ Interface ។ Interface នីមួយៗមានតម្លៃពិតប្រាកដហើយផ្លូវដែលមានតម្លៃទាបជាងគេបំផុតត្រូវបានប្រើ ។ នេះគឺជាតម្លៃរបស់ interfaces:

10 Mbit = Cost 100

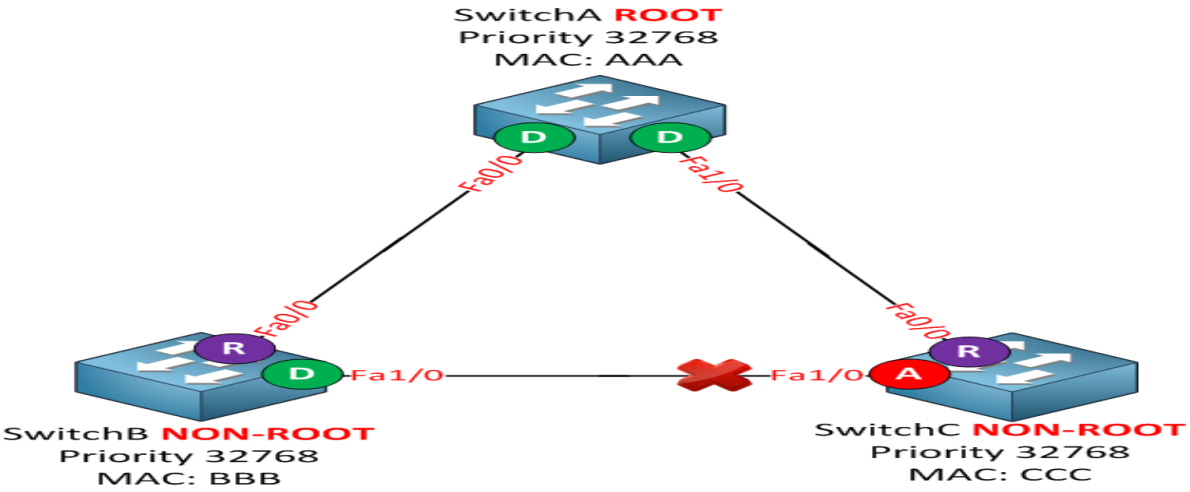
100 Mbit = Cost 19

1000 Mbit = Cost 4

ជាការល្អណាស់ដែលយើងបានកំនត់យក Ports នៅលើ root bridge ហើយនិង root ports នៅលើ non-root bridges។ យើងនៅតែមាន loop ដដែល។ ទោះបីជាយ៉ាងណាក៏ដោយយើងត្រូវតែបិទចោល Port មួយរវាង Switch B និង C ដើម្បីបំបែក loop ចោល។ ដូច្នេះតើ Port មួយណាដែលនិងត្រូវបិទចោល (shutdown)? Port នៅលើ SwitchB ឬ Switch C? សូមពិនិត្យម្តងទៀតទៅលើ bridge ID ដែលល្អប្រសើរជាងគេ:

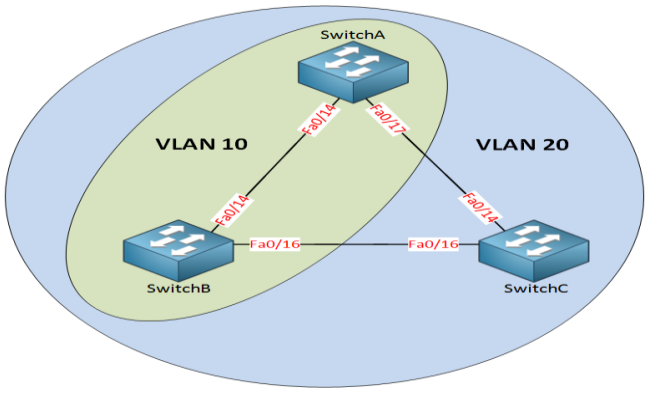
Bridge ID = MAC address + Priority.

តម្លៃទាបជាងគេគឺជាប្រសើរជាងគេ។ Switches ទាំងពីរមានអាទិភាពដូចគ្នា។ ប៉ុន្តែ MAC address នៃ switch B គឺទាបជាងគេ។ នេះមានន័យថា switch B និងឈ្នះ។ Switch C គឺជាអ្នកចាញ់ដែលមានន័យថា port របស់វាត្រូវបានបិទ។ ជាលទ្ធផលគឺ loop ត្រូវបានបំបែកចោល។



Per VLAN Spanning Tree (PVST)

ដោយសារតែអ្នកកំពុងតែអានទៅលើ “មូលដ្ឋានគ្រឹះ” ដែល spanning-tree ធ្វើការ។ សូមចាប់ផ្តើមជាមួយរូបភាពខាងក្រោម:



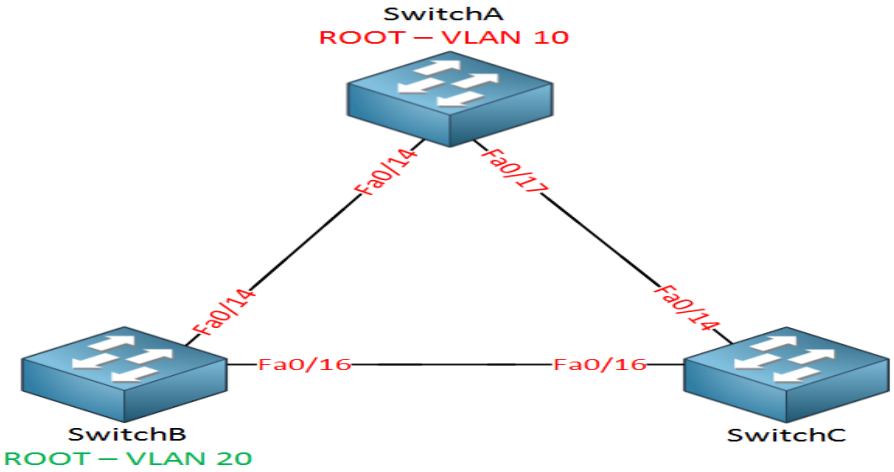
VLAN 10 ត្រូវបាន Configure នៅលើ Switch A និង Switch B

VLAN 20 ត្រូវបាន Configure នៅលើ Switch A , Switch B និង Switch C

តើយើងត្រូវបាន loop នៅក្នុង VLAN 10 ឬទេ? តើយ៉ាងណាដែរចំពោះ VLAN 20?

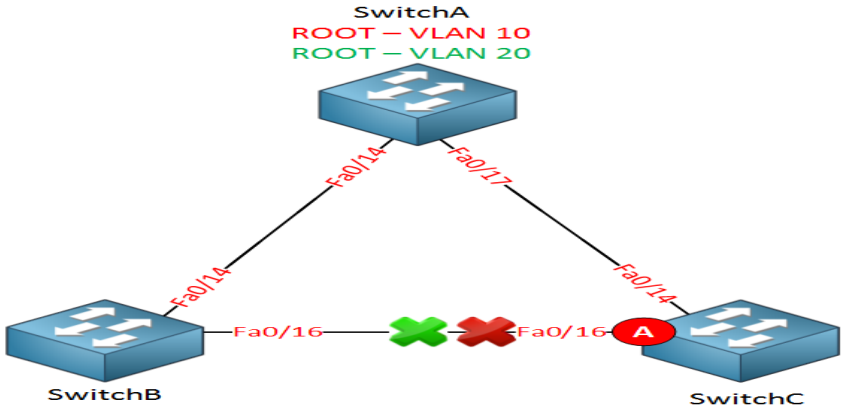
មានភាពខុសគ្នាយ៉ាងខ្លាំងរវាងរូបរាងខាងក្រៅនិងរូបរាងខាងក្នុង? គ្មាន loop កើតមានឡើងទេនៅក្នុង VLAN 10 ពីព្រោះថាវាដំណើរការតែនៅលើការភ្ជាប់ពី SwitchA និង SwitchB។ យើងគ្មាន loop នោះទេជាមួយ VLAN 20 ។

តើ Spanning-tree ដំណើរការជាមួយ loop របៀបណា? យើងគ្រាន់តែគណនាពី Spanning-tree ខុសគ្នារវាង VLAN នីមួយៗ។ ចំពោះ Version ចាស់របស់ spanning-tree គេហៅថា CST (Common Spanning-Tree) ហើយត្រូវបានកំនត់នៅក្នុងស្តង់ដា 802.1D។ វាគណនា single spanning-tree សម្រាប់ VLANs ទាំងអស់។ ចំពោះ Version នៃ spanning-tree មួយទៀតគឺអាចគណនាចំពោះ topology សម្រាប់ VLAN នីមួយៗ។ Version នេះគេហៅថា PVST(Per VLAN Spanning-Tree)ហើយវាមានជា default នៅលើ Cisco switches ។



ដូចអ្នកបានឃើញខាងលើស្រាប់យើងមាន root bridges ពីរ។ បើយើងប្រើ PVST យើងអាចបង្កើតនូវ root bridge ផ្សេងគ្នាមួយសម្រាប់ VLAN នីមួយៗបើយើងចង់បានវា។ SwitchA អាចជា root bridge សម្រាប់ VLAN 10 ហើយ SwitchB អាចជា root bridge សម្រាប់ VLAN 20។ ហេតុអ្វីបានជាអ្នកចង់ធ្វើដូច្នេះ?

នេះជាឧទាហរណ៍



បើអ្នកចង់យក Switch មួយជា root bridge សម្រាប់ VLANs ទាំងពីរ នោះ Interface មួយនិងត្រូវបិទចោលសម្រាប់ VLANs ទាំងពីរ។ នៅក្នុងឧទាហរណ៍ខាងលើ Switch A គឺជា root bridge សម្រាប់ VLAN 10 និង 20 ហើយជាលទ្ធផលគឺ fa0/16 interface នៅលើ SwitchC ត្រូវបានបិទចោលសម្រាប់ VLANs ទាំងពីរ។ គ្មានចរាចរណ៍ណានិងត្រូវបានបញ្ជូននៅលើ fa0/16 interface នោះទេ។

សូមគិតទៅលើ 10 Gigabit interfaces ។ បើ interface ដ៏មានតម្លៃនេះមិនអាចអ្វីបានត្រឹមត្រូវនោះវាគួរឲ្យខ្មាសអៀនណាស់។

បើអ្នកជ្រើសរើសយក Switch មួយទៀតជា root bridge សម្រាប់ VLAN 20 យើងនិងឃើញនូវលទ្ធផលខុសគ្នា។

៩-២-Spanning Tree Port States

បើអ្នកធ្លាប់បានប្រើ Cisco Switches ខ្លះពីមុនមក អ្នកអាចមានការកត់ចំណាំថានៅគ្រប់ពេលដែលអ្នកដោយខ្សែ នោះ interface ខាងលើមានព័ណ៌បៃតងហើយក្រោយបន្តិចមកបានក្លាយជាខៀវវិញ។ អ្វីដែលកំពុងតែកើតមានឡើងគឺ Spanning tree កំពុងតែកំនត់ពីលក្ខណៈនៃ interface ។

វាកើតមានឡើងភ្លាមនៅពេលអ្នកដោតខ្សែចូល:

Listening state:មានតែ root ឬ designated port និងស្ថិតនៅក្នុង listening state ។ ចំពោះ non-designated port និងស្ថិតនៅក្នុងលក្ខណៈជា blocking state។ គ្មានការបញ្ជូនទិន្នន័យណាមួយប្រព្រឹត្តិទៅនោះទេក្នុងរយៈពេល ១៥នាទី។ ក្រោយពីស្ថិតនៅក្នុងការស្តាប់ វាស្ថិតនៅក្នុង learning state។

Learning state: នៅពេលនេះ interface និងដំណើរការចំពោះ Ethernet frames ដោយពិនិត្យទៅលើ MAC address របស់ប្រភពដើមដើម្បីបំពេញក្នុងតារាងនៃ mac-address-table។ ទោះបីជាយ៉ាងណាក៏ដោយ Ethernet frames មិនត្រូវបានបញ្ជូនទៅកាន់គោលដៅនោះទេ។ វាចំណាយ 15 វិនាទីដើម្បីឈានចូលទៅ forwarding state។

Forwarding state: នេះគឺជាលក្ខណៈចុងក្រោយនៃ Interface ហើយជាចុងក្រោយ interface និងបញ្ជូន Ethernet frame ។ ដូច្នេះវាមានធ្វើការបញ្ជូនទិន្នន័យ។

នៅពេលដែល Port មិនមែនជា designated ឬ root port វានិងស្ថិតនៅក្នុង blocking mode ។ នេះមានន័យថាវាចំណាយ 30 វិនាទីដើម្បីបំលាស់ពីការស្តាប់មកជាការបញ្ជូន ។ វាមិនលឿននោះទេ។ វាកើតមានឡើងនៅគ្រប់ interfaces ទាំងអស់នៅលើ Switch ។

នៅពេល interface មួយស្ថិតនៅក្នុង blocking mode ហើយ topology បានប្តូរ។ វាអាច Interface ដែលកំពុងស្ថិតនៅក្នុង blocking mode ដើម្បីឈានមក forwarding state។ នៅពេលវាស្ថិតនៅក្នុងករណីនេះ blocking mode និងស្ថិតនៅរយៈពេល ២០វិនាទីមុនពេលវាឈានមក listening state ។ នេះមានន័យថាវា

ចំណាយ 20 (blocking) + 15 (listening) + 15 (learning) = 50 វិនាទីមុនពេលដែល Interface ស្ថិតនៅក្នុង forwarding state ។

ខាងក្រោមនេះគឺជាលក្ខណៈនៃ Port :

State	Forward Frames	Learn MAC Addresses	Duration
Blocking	No	No	20 seconds
Listening	No	No	15 seconds
Learning	No	Yes	15 seconds
Forwarding	Yes	Yes	–

តើមានលក្ខណៈយ៉ាងណាចំពោះ Cisco Switch ពិតប្រាកដ? ឥឡូវនេះសូមបង្ហាញឧទាហរណ៍នៃ interface មួយដែលត្រូវបានភ្ជាប់ជាមួយ Router មួយ។ យើងដកហើយដោយខ្សែវិញ(កុំប្រើ a"shut" និង "no shut") ហើយជាដំបូងយើងប្រើបញ្ជាដូចខាងក្រោម:

```
SW1#show spanning-tree vlan

VLAN0001

  Spanning tree enabled protocol ieee

  Root ID    Priority    32769

             Address     0019.569d.5700

             This bridge is the root

             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  ( priority 32768 sys-id-ext 1 )

             Address     0019.569d.5700

             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

             Aging Time 300

       Interface          Role Sts Cost      Prio.Nbr Type

------------------- ---- --- --------- -------- --------------------------------
```

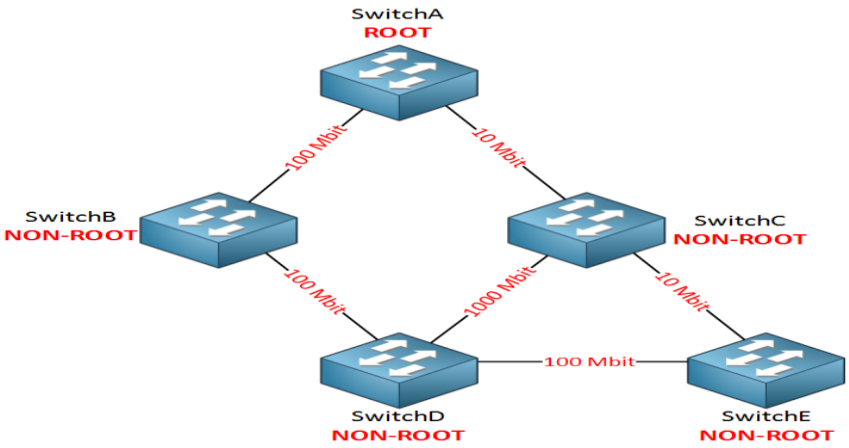
```
Fa0/1            Desg LIS 19       128.4    P2p
```

អ្នកឃើញថាតួមានទីរបស់ Port ត្រូវបានកំនត់ហើយលក្ខណៈរបស់វាគឺកំពុងស្តាប់។ ក្រោយរយៈពេល 15 វិនាទី វានិងមានលក្ខណៈដូចខាងក្រោមនេះ:

```
SW1#show spanning-tree vlan 1

VLAN0001

  Spanning tree enabled protocol ieee

  Root ID    Priority    32769

             Address     0019.569d.5700

             This bridge is the root

             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  ( priority 32768 sys-id-ext 1 )

             Address     0019.569d.5700

             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

             Aging Time 300

Interface          Role Sts Cost      Prio.Nbr Type

------------------- ---- --- --------- -------- --------------------------------

Fa0/1              Desg LRN 19        128.4    P2p
```

៩-៣-Spanning-Tree Cost Calculation

Non-root bridges ត្រូវការរកនូវផ្លូវបញ្ជូនដែលខ្លីបំផុតមកកាន់ root bridge ។ តើមានអ្វីកើតមានឡើងបើយើងមានការបញ្ជូនគ្នានៃ Ethernet និង FastEthernet និង Gigabit? នេះគឺជា topology ដែលយើងនិងពន្យល់ពីការគណនាតម្លៃនៃ spanning-tree :



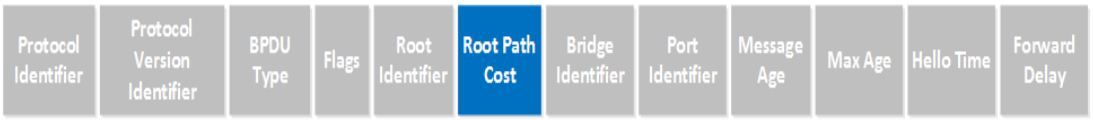
នៅក្នុងរូបភាពខាងលើ ណេតវើកដ៏ធំមួយដែលមាន Switches ច្រើន។ អ្នកក៏អាចឃើញថាមាន Interfaces ជាច្រើនប្រភេទមានដូចជា: Ethernet (10 Mbit), FastEthernet (100Mbit) និង Gigabit (1000Mbit)។ Switch A នៅខាងលើបំផុតគឺជា root bridge ។ ដូច្នេះគ្រប់ Switches ផ្សេងទៀតគឺជា non-root ហើយត្រូវការរកនូវផ្លូវដ៏ខ្លីបំផុតមកកាន់ root bridge។

Bandwidth Cost

10 Mbit	100
100 Mbit	19
1000 Mbit	4

Spanning-tree ប្រើ cost ដើម្បីកំនត់ផ្លូវដែលខ្លីបំផុតមកកាន់ root bridge ។ Interface ដែលមានតម្លៃខ្លីជាងគេគឺជាមានតម្លៃខ្ពស់ជាងគេ។ ផ្លូវដែលមានតម្លៃទាបជាងគេបំផុតនិងត្រូវបានប្រើដើម្បីឈានមកកាន់ root bridge ។ នេះគឺជាកន្លែងដែលអ្នកអាចរកតម្លៃបាន:

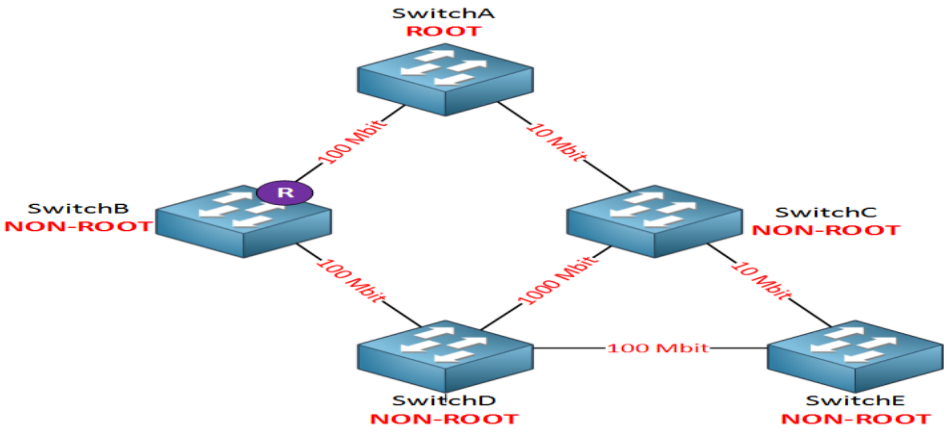
BPDU



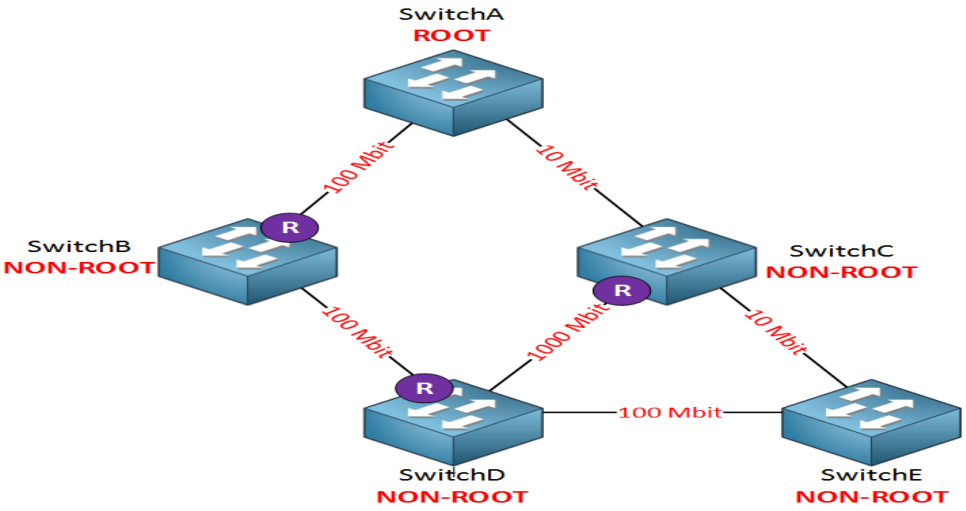
នៅក្នុង BPDU អ្នកអាចឃើញផ្នែកមួយហៅថា root path cost។ នេះគឺជាកន្លែងដែល Switch នីមួយៗ និងបញ្ចូលតម្លៃនៃផ្លូវដែលខ្លីជាងគេបំផុតរបស់វាមកកាន់ root bridge។ នៅពេលដែល Switches បានរកឃើញ Switch ដែលត្រូវបានប្រកាសឲ្យដឹងថាជា root bridge វានិងពិនិត្យមើលផ្លូវដែលខ្លីជាងគេបំផុតដើម្បីទៅដល់វា។ BPDUs និងធ្វើដំណើរពី root bridge ចុះមកកាន់គ្រប់ Switches ទាំងអស់។

OSPF (Open Shortest Path First) ក៏ប្រើតម្លៃដើម្បីគណនាអំពីផ្លូវបញ្ជូនដែលខ្លីជាងគេបំផុតមកកាន់គោលដៅរបស់វា។ ទាំង spanning-tree និង OSPF ប្រើតម្លៃដើម្បីស្វែងផ្លូវដែលខ្លីជាងគេបំផុត។ OSPF បង្កើត topology database (LSDB) មួយ។ ដូច្នេះគ្រប់ Routers ទាំងអស់ដឹងយ៉ាងពិតប្រាកដពីអ្វីដែល ណេតវើកមាន។

Spanning-tree គឺមិនឆ្លាត។ Switches មិនបានដឹងពីរូបរាងនៃ Topology បែបណានោះទេ។ BPDUs និងបញ្ជូនចេញពី root bridge ចុះមកមាន់ switches ទាំងអស់។ Switches និងធ្វើការសម្រេចចិត្តពឹងផ្អែកទៅលើ BPDUs ដែលវាបានទទួល។ នេះគឺជាឧទាហរណ៍ចំពោះតម្លៃនៃ spanning tree ខុសគ្នាសម្រាប់ Topology របស់យើង:



SwitchB និងប្រើខ្សែដែលភ្ជាប់ដោយផ្ទាល់មកកាន់ SwitchA ជា root port របស់វាពីព្រោះថាវាជា Interface ដែលមានល្បឿន 100 Mbit ហើយមានតម្លៃស្មើនិង 19។ វានិងបញ្ជូន BPDUs មកកាន់ Switch D ។ នៅក្នុងផ្នែកនៃ root path របស់ BPDU អ្នកនិងឃើញតម្លៃ 19។ SwitchC ក៏ទទួលបាននូវ BPDUs ពី SwitchA។ ដូច្នេះវាជ្រើសរើសយក Interface ដែលមានល្បឿន 10 Mbit interface ដែលជា root port។



នេះគឺជារូបភាពដែលត្រូវការនូវការពន្យល់ខ្លះមានដូចខាងក្រោមនេះ:

SwitchC និងទទួល BPDUs តាមរយៈ interface ដែលមានតម្លៃ 100 មានល្បឿន 10 Mbit នឹង Interfce ដែលមានល្បឿន 1000 Mbit interface (មានតម្លៃ 4)។ វាប្រើ Interface ដែលមានល្បឿន 1000 Mbit ជា root port របស់វា។

SwitchC និងបញ្ជូន BPDUs មកកាន់ SwitchD។ ផ្នែក root path cost គឺ 100។

SwitchD ទទួលនូវ BPDU មួយពី SwitchB ជាមួយ root path មួយដែលមានតម្លៃស្មើនឹង 19

SwitchD ទទួលនូវ BPDU មួយពី SwitchC ដែលមាន root path មួយមានតម្លៃស្មើ 100

ផ្លូវឆ្លងកាត់តាម Switch B គឺខ្ពីជាងគេ។ ដូច្នេះវានិងក្លាយជា root port សម្រាប់ SwitchD

SwitchD និងបញ្ជូន BPDUs មកកាន់ SwitchC និង SwitchE។ នៅក្នុងផ្នែកនៃ root path cost របស់ BPDU យើងនិងរកឲ្យតម្លៃ 38 (តម្លៃនៃ root path ស្មើ 19 + តម្លៃនៃ Interface ផ្ទាល់ខ្លួនរបស់វាស្មើនឹង 19)

SwitchC និងបញ្ជូនបន្តនូវ BPDUs ឆ្ពោះទៅ SwitchE នឹងបញ្ចូលតម្លៃស្មើនឹង 42 នៅក្នុង root path cost field (19 + 19 + 4) ។

៩-៤-Cisco Portfast Configuration

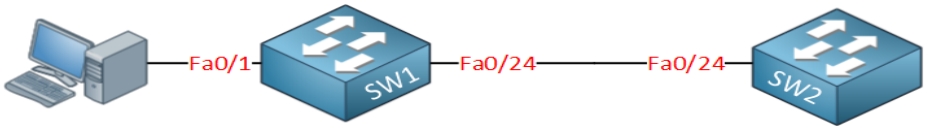
Portfast គឺជាដំណោះស្រាយរបស់ Cisco ដែលទាក់ទងជាមួយការផ្លាស់ប្តូរនៃ spanning-tree topology ។

Portfast ធ្វើកិច្ចការដូចខាងក្រោម:

Interfaces ដែលមាន portfast បើកនិងស្ថិតនៅក្នុង forwarding mode ភ្លាម។ interface និងរំលងចោលចំពោះ listening និង learning state ។

switch នឹងមិនបង្កើតនូវការសម្គាល់ពីការផ្លាស់ប្តូរនៃ topology សម្រាប់ interface ដែលបានបើក portfast ។ ជាការល្អដែលអ្នកត្រូវតែបើក portfast នៅលើ interfaces ដែលត្រូវបានភ្ជាប់ជាមួយ hosts ពីព្រោះថា Interfaces ទាំងនេះបិទ បើកគ្រប់ពេលវេលា។ សូមកុំបើក portfast នៅលើ interface មួយដែលភ្ជាប់មកកាន់ Hub ឬ Switch ។

សូមពិនិត្យមើលពីភាពខុសគ្នានៃ Interface មួយដែលមាន portfast និងគ្មាន portfast។ យើងនិងប្រើ Topology ដូចខាងក្រោមនេះ:



យើងមាន Switches ពីរនិងមាន host មួយដែលភ្ជាប់ជាមួយ Sw1 ។ ហេតុផលតែមួយគត់ដែលយើងប្រើ Switches ពីរគឺ SW1 មាន Switch មួយទៀតដែលវាអាចបញ្ជូននូវការកត់សម្គាល់នៃការផ្លាស់ប្តូរ topology ។ សូមពិនិត្យមើលគ្មាន portfast

Portfast disabled

វាជាការដែលគួរឲ្យចាប់អារម្មណ៍ដែលយើងនិង debug នៅលើ SW1:

SW1#debug spanning-tree events

Spanning Tree event debugging is on

Once I plug in the cable to connect the host to SW1 this is what happens:

SW1#

STP: VLAN0001 Fa0/1 -> listening

STP: VLAN0001 Fa0/1 -> learning

STP: VLAN0001 Fa0/1 -> forwarding

នេះគឺជាលក្ខណៈធម្មតានៃ spanning-tree វាដំណើរការតាម listening និង learning states ហើយចុងក្រោយគឺ forwarding។

រាល់ពេលដែលយើងដកខ្សែ spanning-tree និងបង្កើតនូវការសម្គាល់ពីការផ្លាស់ប្តូរនៃ Topology។ មានបញ្ជាដ៏ល្អដែលអ្នកអាចត្រួតពិនិត្យទៅលើវា:

SW1#show spanning-tree detail

VLAN0001 is executing the ieee compatible Spanning Tree protocol

Bridge Identifier has priority 32768, sysid 1, address 0019.569d.5700

Configured hello time 2, max age 20, forward delay 15

Current root has priority 32769, address 0011.bb0b.3600

Root port is 26 (FastEthernet0/24), cost of root path is 19

Topology change flag not set, detected flag not set

Number of topology changes 5 last change occurred 00:02:09 ago

from FastEthernet0/1

Times: hold 1, topology change 35, notification 2

hello 2, max age 20, forward delay 15

Timers: hello 0, topology change 0, notification 0, aging 300

ដូចបានបង្ហាញខាងលើមានការផ្លាស់ប្តូរ 5 topology នៅលើ VLAN 1។ នៅពេលដកខ្សែចំពោះ Host យើងឃើញថាមានអ្វីកើតឡើង:

SW1#

STP: VLAN0001 sent Topology Change Notice on Fa0/24

Spanning-tree និងបញ្ជូននូវការសម្គាល់ពីការប្តូរនៃ Topology នៅលើ interfaces មកកាន់ SW2និងជួប:

SW1#show spanning-tree detail | include changes

Number of topology changes 6 last change occurred 00:01:12 ago

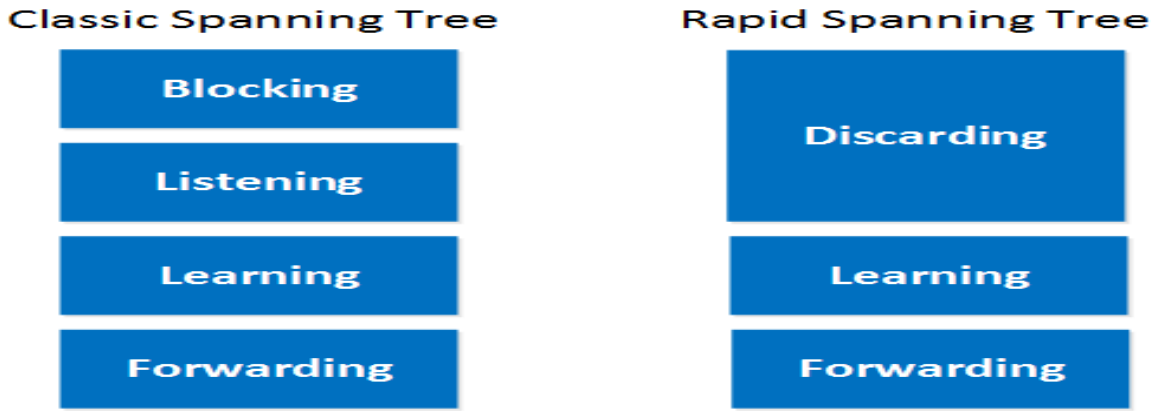
សរុបមកវិញ គ្រប់ពេលដែលមានការដកខ្សែ Switch និងបង្កើតនូវ TCN ។

៩-៤-១-Rapid Spanning-Tree (RSTP)

សព្វថ្ងៃនេះយើងប្រើ Routing កាន់តែច្រើនឡើងនៅក្នុងប្រព័ន្ធណេតវើក ។ Routing Protocols មានដូចជា OSPF និង EIGRP ដែលជា Protocols ដែលមានល្បឿនលឿនជាង spanning-tree នៅពេលដែលវាមានការផ្លាស់ប្តូរនៃរូបរាងនៅក្នុង ណេតវើក ។ ដើម្បីបំពេញទៅតាមតម្រូវការនៃ Spanning-tree ឲ្យ មានល្បឿនលឿននោះគេបង្កើតនូវrapid spanning-tree ។

Rapid spanning-tree មិនមែនជាការធ្វើបដិវត្តនៃ Spanning-tree ចាស់នោះទេ ។ ប៉ុន្តែវាជាការធ្វើឲ្យល្អប្រសើរ ។ អ្វីដែលបានប្តូរចំពោះវានោះគឺបង្កើនល្បឿននៃដំណើរការ ការ Configure មានភាពវៃឆ្លាត ។

សូមសិក្សាពី rapid spanning-tree យើងនឹងឃើញពីភាពខុសគ្នាជាមួយ classic spanning-tree ។ សូមមើលពីរូបភាពខាងក្រោម:



សូមចាំពីលក្ខណៈ Port របស់ spanning-tree ។ យើងមានលក្ខណៈរបស់វាដូចជា blocking,listening, learning និង forwarding port ។ នេះគឺជាភាពខុសគ្នាទី១រវាង spanning-treeនិងrapid spanning-tree ។

ចំពោះ Rapid spanning-tree port មានលក្ខណៈបីយ៉ាងគឺ:

Discarding

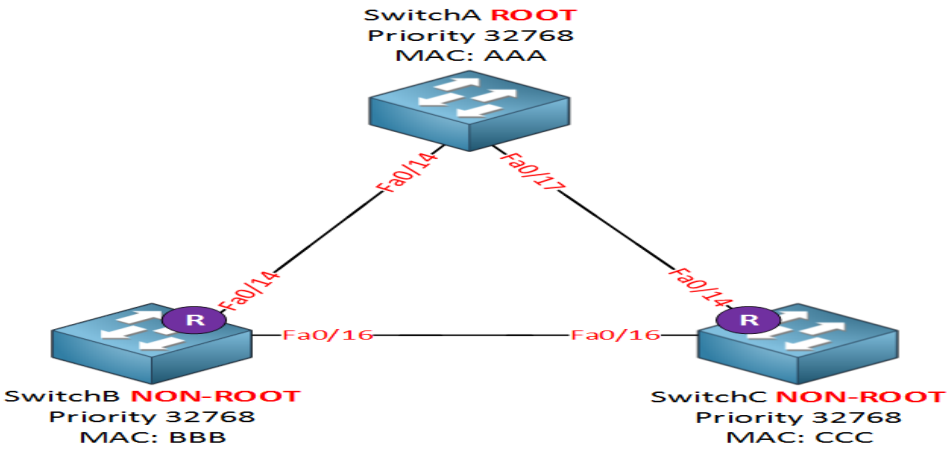
Learning

Forwarding

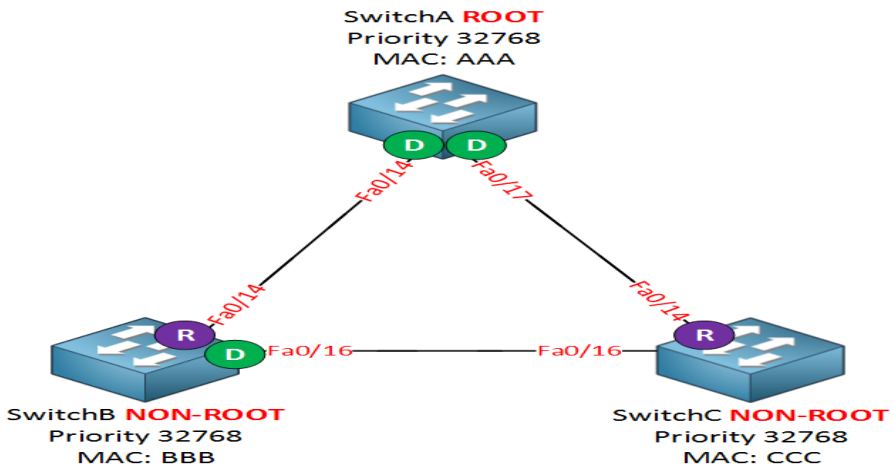
យើងមានដឹងរួចហើយអំពី learning និង forwarding។ ចំណែកឯ discarding គឺជាលក្ខណៈនៃ Port ថ្មីមួយទៀត។ វាគឺជាការបញ្ចូលគ្នានៃ blocking និង listening port state។ ខាងក្រោមនេះគឺជាការបង្ហាញពីលក្ខណៈរបស់វា:

Classic Spanning-Tree	Rapid Spanning-Tree	Port active in topology ?	Learns MAC addresses ?
Blocking	Discarding	No	No
Listening	Discarding	Yes	No
Learning	Learning	Yes	Yes
Forwarding	Forwarding	Yes	Yes

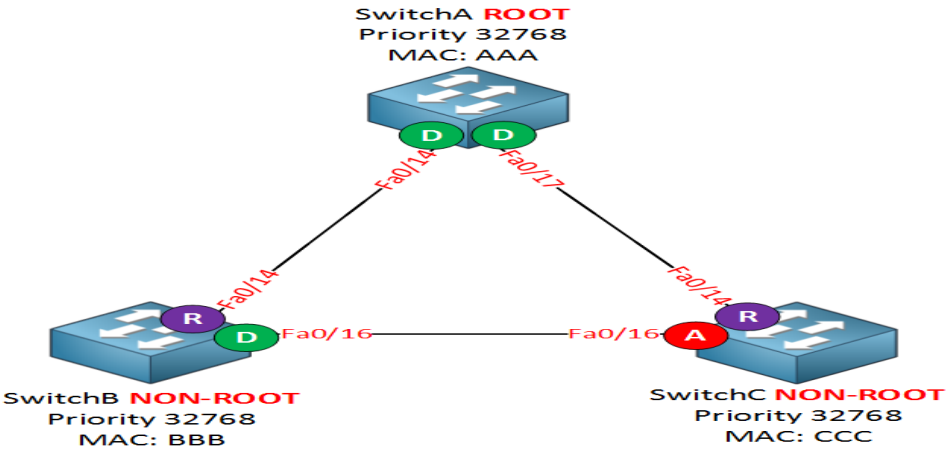
តើអ្នកបានចាំទេចំពោះតួនាទីទាំងអស់របស់ Ports ដែល Spanning-tree មាន? សូមរម្លឹកឡើងវិញបន្តិចហើយយើងបង្ហាញពីភាពខុសគ្នាសម្រាប់ rapid spanning-tree:



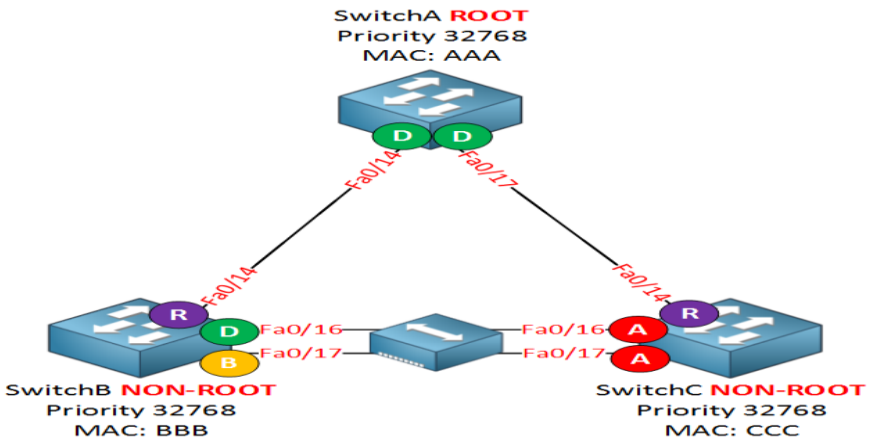
ចំពោះ Switch ដែលមាន Bridge ID ល្អប្រសើរ (priority + MAC address) ក្លាយជា root bridge។ ចំពោះ Switches ផ្សេងៗទៀត (non-root) ត្រូវតែរកផ្លូវដែលមានតម្លៃខ្ពីជាងគេបំផុតមកកាន់ root bridge។ នេះគេហៅថា root port។ គ្មានអ្វីថ្មីនោះទេ។ ជំហានបន្ទាប់គឺត្រូវជ្រើសរើសយក designated ports:



នៅលើ Segment នីមួយៗអាចមាន designated port តែមួយគត់ឬវាបញ្ចប់ជាមួយ loop មួយ។ Port នឹងក្លាយជា designated port បើវាអាចបញ្ជូននូវ BPDU ដែលល្អប្រសើរ។ Switch A ជា Root bridge មួយនិងមាន Ports ដ៏ល្អប្រសើរ។ ដូច្នេះគ្រប់ Interfaces និងត្រូវបានកំនត់។ fa0/16 interface នៅលើ SwitchB ក្លាយជា designated port នៅក្នុងឧទាហរណ៍ពីព្រោះថាវាមាន Bridge ID ដ៏ប្រសើរជាង SwitchC ។ វានៅតែគ្មានអ្វីថ្មីនោះទេ បើប្រៀបធៀបជាមួយ classic spanning-tree។ interfaces នៅខាងឆ្វេងនិងត្រូវបិទចោល។



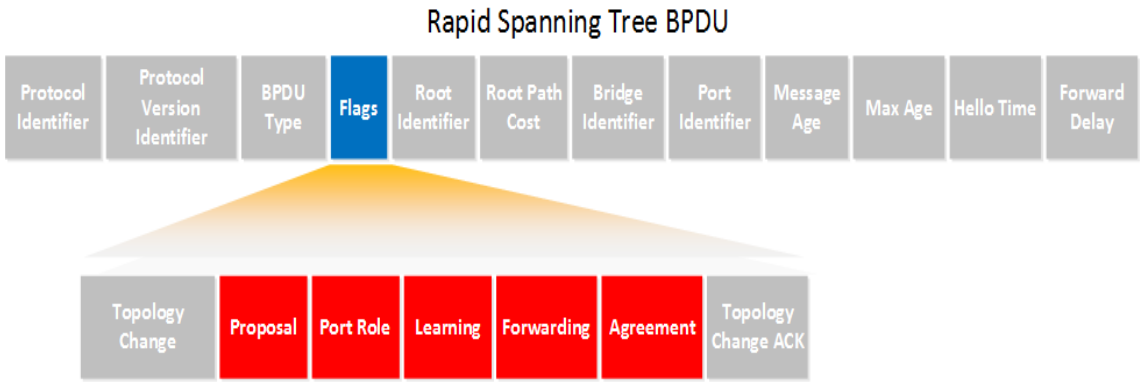
SwitchC ទទួលនូវ BPDU ល្អប្រសើរនៅលើ fa0/16 interface ពី SwitchB ហើយវានិងត្រូវបិទចោល។ នេះគឺជា Port ឆ្លាស់ហើយវានៅតែដូចគ្នាចំពោះ rapid spanning-tree។ សូមមើលពីឧទាហរណ៍ថ្មីមួយ ទៀតជាមួយ port state សម្រាប់ rapid spanning-tree:



នេះគឺជា Port ថ្មីមួយសម្រាប់អ្នក។ សូមពិនិត្យមើលទៅលើ Interface fa0/17 នៃ SwitchB។ គេហៅវាថាជា backup port ហើយវាថ្មីសម្រាប់ rapid spanning-tree។ អ្នកប្រហែលជាមិនបានឃើញ port នេះនៅលើ production ណេតវើកនោះទេ។ រវាង SwitchB និង SwitchC គេបានបន្ថែម Hub មួយ។ តាមធម្មតា(វាគ្មាន Hub នៅចន្លោះនោះទេ) ចំពោះ fa0/16 និង fa0/17 គឺជា designated ports។

ដោយសារតែ Hub នោះ fa0/16 និង fa0/17 interface នៅលើ SwitchB គឺស្ថិតនៅក្នុង collision domain ដូចគ្នា។ Fa0/16 និងត្រូវបានជ្រើសរើសជា designated port ហើយ fa0/17 និងក្លាយជា backup port សម្រាប់ fa0/16 interface។ ហេតុផលដែល SwitchB ចាត់ទុក interface fa0/17 ជា backup port ដោយសារតែវាទទួល

បាននូវ BPDUs ផ្ទាល់ខ្លួនរបស់វានៅលើ fa0/16 និង fa0/17 interfaces ហើយយល់ថាវាមានការភ្ជាប់ពីរមកកាន់ segment តែមួយ។ បើអ្នកយក Hub ចេញ នោះ fa0/16 និង fa0/17 នឹងក្លាយជា designated ports ដូចជា classic spanning-tree ដែរ។ សូមពិនិត្យមើលអ្វីដែលខុសគ្នាគឺ BPDU ។



BPDU គឺខុសគ្នាសម្រាប់ rapid spanning-tree។ នៅក្នុង classic spanning-tree គឺ flags field មានតែ 2 bits ដែលត្រូវប្រើ:

Topology change.

Topology change acknowledgment.

ឥឡូវនេះគឺ bits ទាំងអស់នៃ flag field ត្រូវបានប្រើ។ តួនាទីនៃ Port ដែនកំនត់ពីប្រភពដើមនៃ BPDU នឹងត្រូវបានបន្ថែមដោយប្រើ port role field ដែលមាន options ដូចខាងក្រោម:

Unknown

Alternate / Backup port.

Root port.

Designated port.

BPDU ថ្មីនេះហៅថា version 2 BPDU ។ Switches ដែលកំពុងដំណើរការនៃ version ចាស់របស់ Spanning-tree និងបោះចោលចំពោះ BPDU version ថ្មីនេះ។ នៅក្នុងករណីដែលអ្នកសង្ស័យ rapid spanning-tree និង spanning ចាស់គឺត្រូវគ្នា។ ចំពោះ Rapid spanning-tree មានវិធីសាស្ត្រអាចដំណើរការចំពោះ spanning-tree ចាស់ជាមួយ Switch បាន។

សូមសិក្សាទៅលើកិច្ចការមួយចំនួនដែលបានផ្លាស់ប្តូរ:

BPDUs ឥឡូវនេះត្រូវបានបញ្ជូនគ្រប់ពេលនៃ Hello time ។ មានតែ root bridge តែមួយគត់ដែលបង្កើតនូវ BPDUs នៅក្នុង classic spanning-tree ហើយវាត្រូវបានពន្យាពេលដោយ non-root switches ។ បើវាទទួល

នៅលើ root port របស់វា។ ចំពោះ Rapid spanning-tree ធ្វើការខុសគ្នាគឺគ្រប់ Switches ទាំងអស់បង្កើត BPDUs នៅរៀងរាល់២វិនាទី (hello time)។ វាគឺជា default hello time ប៉ុន្តែយើងបានប្តូរវា។

ចំពោះ classic spanning-tree វិញប្រើការកំនត់ពេលវេលាយូរបំផុតគឺ ២០វិនាទីសម្រាប់ BPDUs ។ មុនពេលវាត្រូវបានបោះចោល។ Rapid spanning-tree ធ្វើការខុសគ្នា។ BPDUs ឥឡូវនេះត្រូវបានប្រើជា keepalive mechanism មួយដែលវាដូចទៅនិងអ្វីដែល routing protocols ដូចជា OSPF ឬ EIGRP បានប្រើ។ បើ Switch មួយបានបាត់ BPDUs ចំនួនបីពី Switch ជិតខាងរបស់វា ។ វាសន្មត់ថាការភ្ជាប់មកកាន់ Switch នេះត្រូវបានផ្តាច់ ហើយវានឹងលុបចោលនូវ MAC addresses ទាំងអស់ភ្លាម។

Rapid spanning tree និងទទួលយក inferior BPDUs។ ចំពោះ classic spanning tree វិញមិនទទួលយកនោះទេ។ នេះគឺជាលក្ខណៈពិសេសនៃ Backbone របស់ classic spanning-tree ដែលមានល្បឿនលឿន។ ល្បឿនផ្ទេរ(convergence time)គឺជាលក្ខណៈពិសេសដ៏សំខាន់បំផុតនៃ rapid spanning tree។ ចំពោះ classic spanning tree ត្រូវតែធ្វើការដោយឆ្លងកាត់លក្ខណៈ listening និង learning មុនពេលវាធ្វើឲ្យ Interface មួយមកស្ថិតនៅក្នុងលក្ខណៈជា forwarding ដែលចំណាយពេល 30 វិនាទីជាមួយការកំនត់ពេលវេលាតាមលំនាំដើម។ classic spanning tree ត្រូវបានពឹងផ្អែកទៅលើ timers។

Rapid spanning មិនបានប្រើ timers ទេដើម្បីសម្រេចថាតើinterface មួយអាចប្តូរមកជាលក្ខណៈ forwarding ឬមិនអាច។ វាប្រើយន្តការចរចារសម្រាប់កិច្ចការនេះ។ តើអ្នកបានចាំចំពោះ portfast ដែរឬទេ?

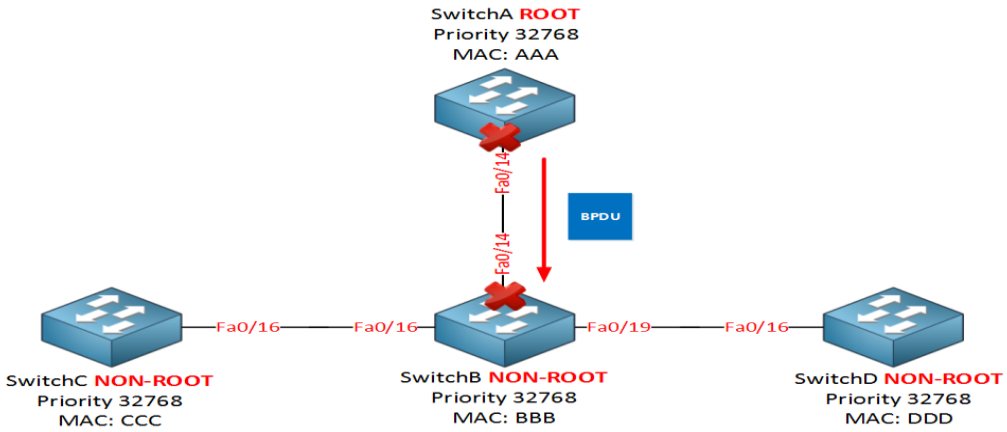
បើអ្នកបើក portfast នៅក្នុងកំឡុងពេលដែលដំណើរការ classic spanning tree វានិងរំលងចំពោះលក្ខណៈ listening និង learning ហើយបញ្ចូល interface នៅក្នុងលក្ខណៈ forwarding ។ ក្រៅពីបំលាស់ interface មកជាលក្ខណៈនៃ forwarding វាក៏មិនបង្កើតនូវការផ្លាស់ប្តូរចំពោះ topology នៅពេល Interface ដំណើរការឬបិទ។

យើងនៅតែប្រើ portfast សម្រាប់ rapid spanning tree ប៉ុន្តែពេលនេះគេហៅថា edge port។Rapid spanning tree អាចដាក់បញ្ចូល interfaces តែនៅក្នុងលក្ខណៈ forwarding បានយ៉ាងលឿននៅលើ edge ports (portfast) ឬ point-to-point interfaces។ មានប្រភេទនៃ interfaces ពីរប្រភេទគឺ

Point-to-point (full duplex)

Shared (half duplex)

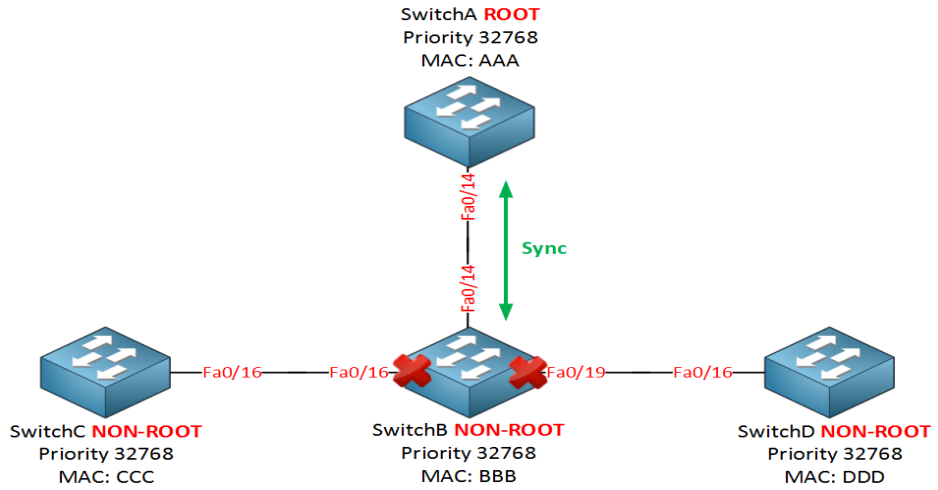
តាមធម្មតាយើងកំពុងប្រើ Switches និងគ្រប់ Interfaces របស់យើងត្រូវបាន Configure ជា full duplex។ rapid spanning tree ឃើញ Interfaces ទាំងនេះជា point-to-point។ បើយើងនិយាយពី Hub វិញចំពោះណេតវើករបស់យើង យើងមាន half duplex ដែលគេបានដឹងថាវាជា shared interface ចំពោះ rapid spanning-tree។ សូមពិនិត្យមើលដោយយកចិត្តទុកដាក់ចំពោះយន្តការចរចារដែលយើងបានពិពណ៌នាខាងលើមក:



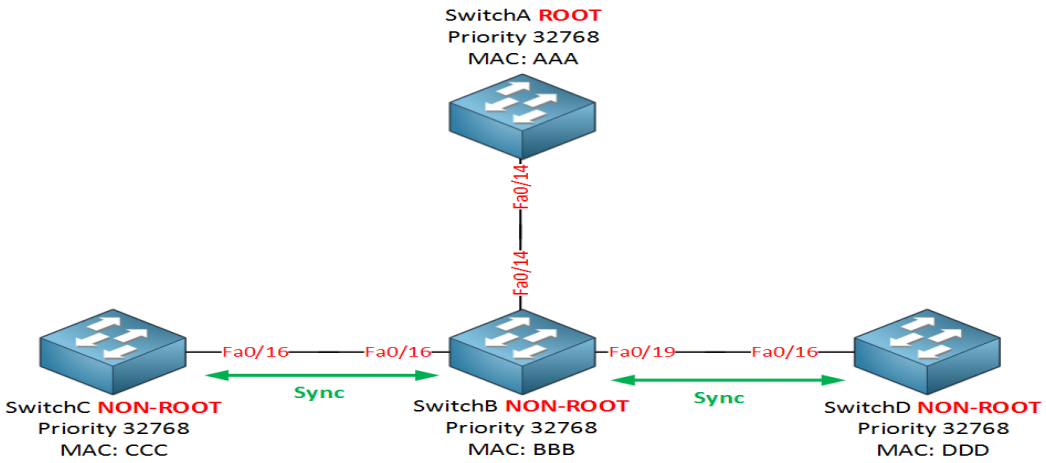
សូមពន្យល់ពីយន្តការដែលធ្វើឲ្យដូចគ្នាចំពោះrapid spanning tree ដោយប្រើរូបភាពខាងលើ។ SwitchA នៅខាងលើគឺជា root bridge មួយ។ SwitchB, SwitchC និង SwitchD គឺជា non-root bridges។

នៅពេលណាដែលការភ្ជាប់រវាង SwitchA និង SwitchB បានបើក interfaces របស់វាស្ថិតនៅក្នុង blocking mode។ SwitchB និងទទួលយកនូវ BPDU មួយពី SwitchA ហើយនិងមានការចរចារគ្នាហៅថា

sync:

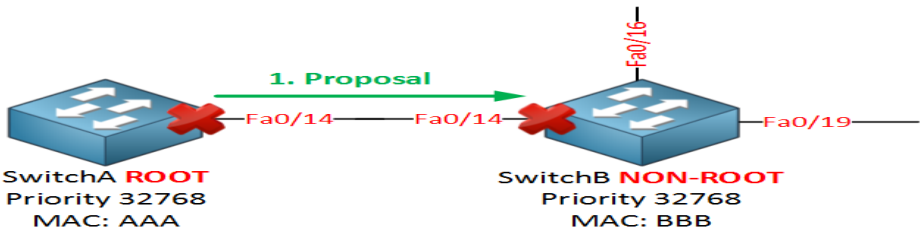


ក្រោយពី SwitchB ទទួលបាននូវ BPDU ពី root brisge ។ វាបិទ(block) ភ្លាមចំពោះ non-edge designated ports។ Non-edge ports គឺជា interfaces ដែលភ្ជាប់មកកាន់ Switches ផ្សេងៗទៀតនៅក្នុងខណៈដែល edge ports គឺជា interfaces ដែលមាន portfast ត្រូវបាន Configure ។ ភ្លាមនោះ SwitchB បិទ(block) ចំពោះ non-edge ports របស់វាដែលភ្ជាប់រវាង SwitchA និង SwitchB ដែលនិងស្ថិតនៅក្នុង forwarding ។ SwitchB និងធ្វើការដូចខាងក្រោមនេះ:



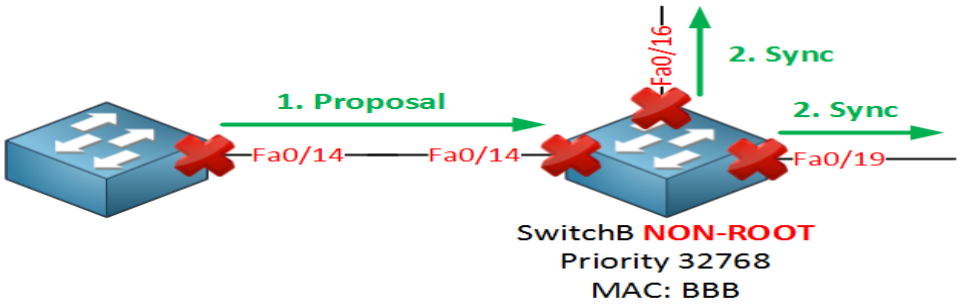
SwitchB ក៏សម្តែងនូវដំណើរការ Sync ជាមួយ SwitchC និង SwitchD ។ ដូច្នេះវាអាចស្ថិតនៅក្នុង forwarding បានយ៉ាងលឿន ។

នៅក្នុងមេរៀននេះគឺថា rapid spanning tree ប្រើយន្តការ sysc ជំនួសឲ្យយន្តការ “timer-based” ដែល classic spanning tree ប្រើ (listening > learning > forwarding) ។ យើងនិងបង្ហាញអ្នកឲ្យបានឃើញអ្វីដែលមាននៅក្នុង Switch ពិតប្រាកដ ។ សូមពិនិត្យមើលចំពោះយន្តការ sync តើមានអ្វីកើតមានឡើងរវាង SwitchA និង SwitchB:

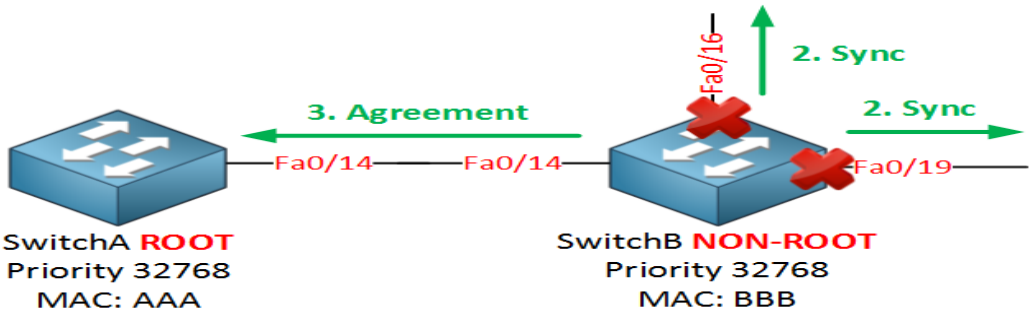


ជាដំបូង Interfaces និងត្រូវបិទ (block) រហូតទាល់តែវាទទួលបាននូវ BPDU មួយពីគ្នាទៅវិញទៅមក ។ នៅពេលនេះ SwitchB និងត្រូវបាន Configure ដែលបញ្ជាក់ឲ្យដឹងថា SwitchA ជា root bridge ពីព្រោះថាវាមាននូវព័ត៌មានអំពី PBDU ដ៏ល្អប្រសើរមួយ ។

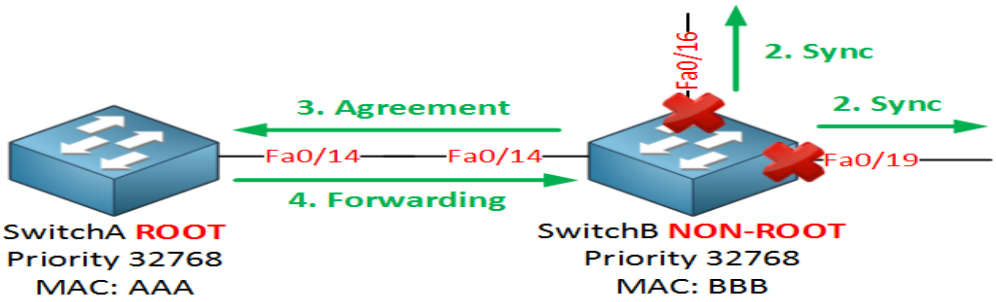
យន្តការ sync និងចាប់ផ្តើមពីព្រោះថា SwitchA និងកំនត់នូវ proposal bit នៅក្នុង flag field នៃ BPDU ។ នៅពេល SwitchB ទទួលនូវសំណើរវាត្រូវតែធ្វើអ្វីមួយ:



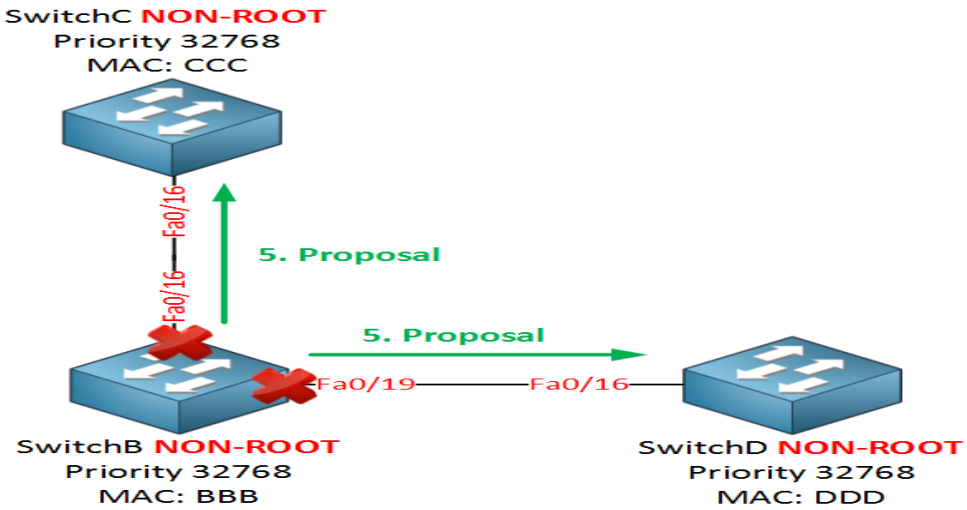
SwitchB និងបិទ (Block) នូវគ្រប់ non-edge interfaces របស់វានិងចាប់ផ្តើមដំណើរការការធ្វើឲ្យដូចគ្នាមកកាន់ SwitchC និង SwitchD។ នៅពេលដែលវាបានសម្រេច នោះ SwitchB និងឲ្យ SwitchA បានដឹងអំពី:



SwitchB ភ្លាមនោះមាន Interfaces របស់វានៅក្នុង sync mode ដែលវានិងអនុញ្ញាតឲ្យ Switch A ដឹងអំពីកិច្ចការនេះដោយបញ្ជូននូវ agreement មួយ។ ការយល់ព្រមនេះគឺជាការចំលងនៃ proposal BPDU ដែលសំណើរ bit ត្រូវបានបិទហើយ bit នៃការយល់ព្រមត្រូវបានបើកឡើង។ fa0/14 interface នៅលើ SwitchB និងស្ថិតនៅក្នុង forwarding mode។ នៅពេលដែល SwitchA ទទួលបាននូវការយល់ព្រមវានិងមានអ្វីកើតមានឡើង:



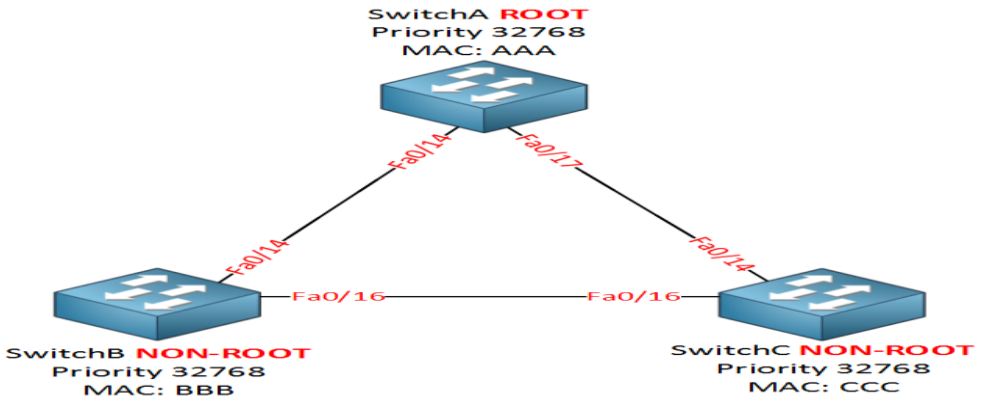
នៅពេលដែល SwitchA ទទួលបាននូវការយល់ព្រមភ្លាមពី SwitchB វានិងដាក់ fa0/14 interface របស់វាស្ថិតនៅក្នុង forwarding mode ភ្លាម។ តើ Interface fa0/16 និង fa0/19 interface នៅលើ SwitchB យ៉ាងណាដែរ?



យន្តការ Sync ដូចគ្នានិងកើតមានឡើងនៅលើ Interfaces ទាំងនេះ។ SwitchB និងបញ្ជូននូវសំណើរមួយនៅលើ Interface fa0/16 និង fa0/19 របស់វាមកកាន់ SwitchC និង SwitchD។ SwitchC និង SwitchD និងទទួលយកនូវការយល់ព្រម។

Rapid Spanning-Tree Configuration

នៅក្នុងមេរៀនមុនបានពន្យល់ពីភាពខុសគ្នានៃ classic and និង spanning-tree ហើយនិងពីរបៀបដែល rapid spanning-tree ធ្វើការ។ សូមពិនិត្យមើលពីការ Configuration ខាងក្រោម។ នេះគឺជា topology ដែលយើងប្រើវា:



ចំពោះ SwitchA និងជា root bridge នៅក្នុងឧទាហរណ៍នេះ។ ជាដំបូងយើងបើក rapid spanning-tree:

SwitchA(config)#spanning-tree mode rapid-pvst

SwitchB(config)#spanning-tree mode rapid-pvst

SwitchC(config)#spanning-tree mode rapid-pvst

បញ្ជាខាងលើគឺប្រើសម្រាប់បើក rapid spanning tree នៅលើ switches។ ការអនុវត្តនៃ rapid spanning treeគឺrapid-pvst។ យើងកំពុងគណនាចំពោះrapid spanning treeសម្រាប់ VLAN នីមួយៗ។

ជាដំបូងយើងបង្ហាញពីរបៀបប្រើយន្តការ sync:

SwitchA(config)#interface fa0/14

SwitchA(config-if)#shutdown

SwitchA(config)#interface f0/17

SwitchA(config-if)#shutdown

ជំហានទី១ ៖យើងនឹងបិទទៅលើ Interfaces នៅលើ Switch A ដើម្បីចាប់ផ្តើម

SwitchA#debug spanning-tree events

Spanning Tree event debugging is on

SwitchB#debug spanning-tree events

Spanning Tree event debugging is on

SwitchC#debug spanning-tree events

Spanning Tree event debugging is on

ជំហានទី២ ៖បើក debug នៅលើ switches ទាំងអស់

SwitchA(config)#interface fa0/14

SwitchA(config-if)#no shutdown

យើងនិងបើក fa0/14 interface នៅលើ SwitchA វិញ ។

SwitchA#

setting bridge id (which=3) prio 4097 prio cfg 4096 sysid 1 (on) id 1001.0011.bb0b.3600

RSTP(1): initializing port Fa0/14

RSTP(1): Fa0/14 is now designated

RSTP(1): transmitting a proposal on Fa0/14

Interface fa0/14 នៅលើ SwitchA និងត្រូវបិទចោល(block) ហើយបញ្ជូននូវសំណើរមកកាន់ SwitchB ។

SwitchB#

RSTP(1): initializing port Fa0/14

RSTP(1): Fa0/14 is now designated

RSTP(1): transmitting a proposal on Fa0/14

RSTP(1): updt roles, received superior bpdu on Fa0/14

RSTP(1): Fa0/14 is now root port

ទោះបីជា SwitchB ជា root bridge ក៏ដោយវានិយាយថាវាបានទទួលនូវ BPDU នៅលើ fa0/14 interface ។វាប្តូរ fa0/14 interface មកជា root port ។

SwitchB# RSTP(1): syncing port Fa0/16

fa0/16 interface នៅលើ SwitchB និងស្ថិតនៅក្នុង sync mode។ នេះគឺជា interface ដែលភ្ជាប់មកកាន់ SwitchC។

SwitchB# RSTP(1): synced Fa0/14

RSTP(1): transmitting an agreement on Fa0/14 as a response to a proposal

SwitchB និងឆ្លើយតបមកកាន់ SwitchA នូវសំណើររបស់វាដោយបញ្ជូននូវការយល់ព្រម។

SwitchA# RSTP(1): received an agreement on Fa0/14

%LINK-3-UPDOWN: Interface FastEthernet0/14, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/14, changed state to up

SwitchA ទទួលការយល់ព្រមពី SwitchB ហើយ interface fa0/14 និងស្ថិតនៅក្នុង forwarding។

SwitchB# RSTP(1): transmitting a proposal on Fa0/16

SwitchB និងបញ្ជូនសំណើរមួយមកកាន់ SwitchC

SwitchC# RSTP(1): transmitting an agreement on Fa0/16 as a response to a proposal

SwitchC និងឆ្លើយតបចំពោះសំណើរនៃ SwitchB ហើយបញ្ជូននូវការយល់ព្រមមួយ

SwitchB# RSTP(1): received an agreement on Fa0/16

%LINK-3-UPDOWN: Interface FastEthernet0/14, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/14, changed state to up

SwitchB ទទួលនូវការយល់ព្រមពី SwitchC ហើយ Interface និងស្ថិតនៅក្នុង forwarding។

ទាំងអស់នេះគឺជា handshakes ហើយ interfaces និងប្តូរមកជា forwarding ដោយគ្មានប្រើ timers ។

SwitchA(config)#interface fa0/17

SwitchA(config-if)#no shutdown

យើងនិងបើក Interface នេះ។ ដូច្នេះការភ្ជាប់ត្រូវបានស្តារឡើងវិញ។

SwitchA#show spanning-tree

VLAN0001

Spanning tree enabled protocol rstp

```
  Root ID    Priority    4097
             Address     0011.bb0b.3600
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    4097   ( priority 4096 sys-id-ext 1 )
             Address     0011.bb0b.3600
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300
Interface          Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/14             Desg FWD 19        128.16   P2p
Fa0/17             Desg FWD 19        128.19   P2p
```

នៅពេលយើងដឹងថា SwitchA ជា root bridge ។ បញ្ជាក៏បញ្ជាក់ថាវាកំពុងដំណើរការ rapid spanning tree

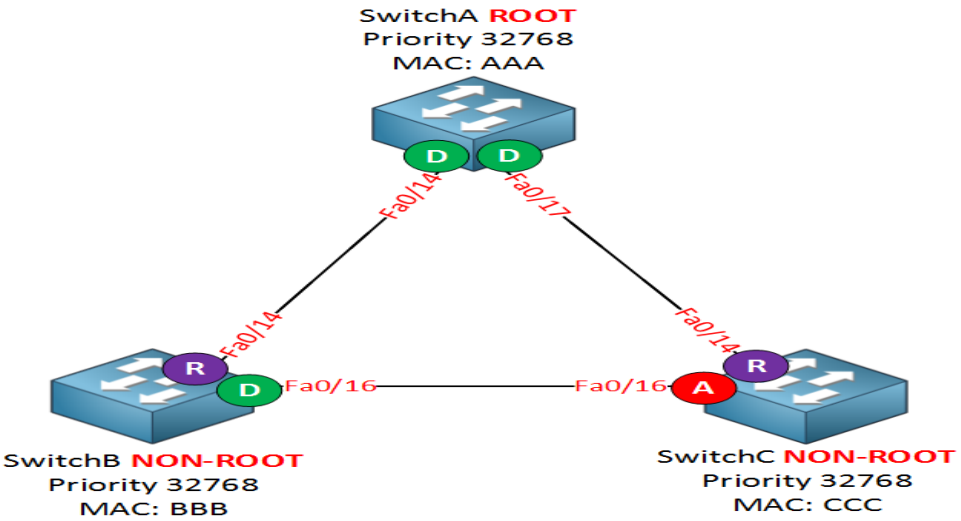
ចំណាំថា៖

ប្រភេទនៃការភ្ជាប់គឺ p2p។ ពីព្រោះថា FastEthernet interfaces គឺជា full duplex តាមលំនាំដើម។ សូមប្រើបញ្ជាខាងក្រោមនៅលើ Switches ពីរ៖

```
SwitchB#show spanning-tree
VLAN0001
  Spanning tree enabled protocol rstp
  Root ID    Priority    4097
             Address     0011.bb0b.3600
             Cost        19
             Port        16 ( FastEthernet0/14 )
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

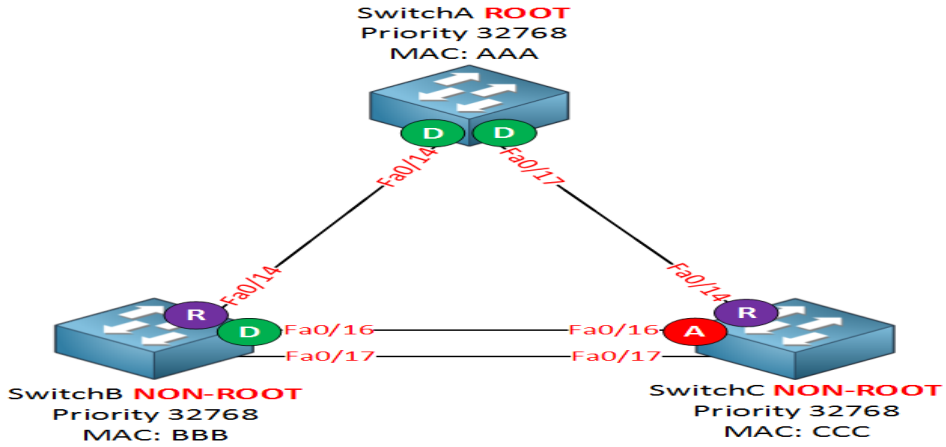
```
  Bridge ID  Priority    8193   ( priority 8192 sys-id-ext 1 )
             Address     0019.569d.5700
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300
Interface          Role Sts Cost      Prio.Nbr Type
------------------- ---- --- --------- -------- --------------------------------
Fa0/14             Root FWD 19        128.16   P2p
Fa0/16             Desg FWD 19        128.18   P2p
SwitchC#show spanning-tree
VLAN0001
  Spanning tree enabled protocol rstp
  Root ID    Priority    4097
             Address     0011.bb0b.3600
             Cost        19
             Port        14 ( FastEthernet0/14 )
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    32769  ( priority 32768 sys-id-ext 1 )
             Address     000f.34ca.1000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/14           Root FWD 19        128.14   P2p
Fa0/16           Altn BLK 19        128.16   P2p
```

នេះគឺ SwitchB និង SwitchC ។ គ្មានអ្វីថ្មីនោះទេ ។ វាមានព័ត៌មានដូចទៅនិង classic spanning tree ដែរ ។



ខាងក្រោមនេះគឺជា topology

ឥឡូវនេះបន្ថែមការភ្ជាប់មួយទៀតរវាង SwitchB និង SwitchC ដើម្បីឲ្យដឹងថាវាមានឥទ្ធិពលទៅលើ topology:



SwitchB#show spanning-tree | begin Interface

Interface Role Sts Cost Prio.Nbr Type

---------- ---- --- --------- -------- --------------------------------

Fa0/14 Root FWD 19 128.16 P2p

Fa0/16 Desg FWD 19 128.18 P2p

Fa0/17 Desg FWD 19 128.19 P2p

SwitchC#show spanning-tree | begin Interface

Interface Role Sts Cost Prio.Nbr Type

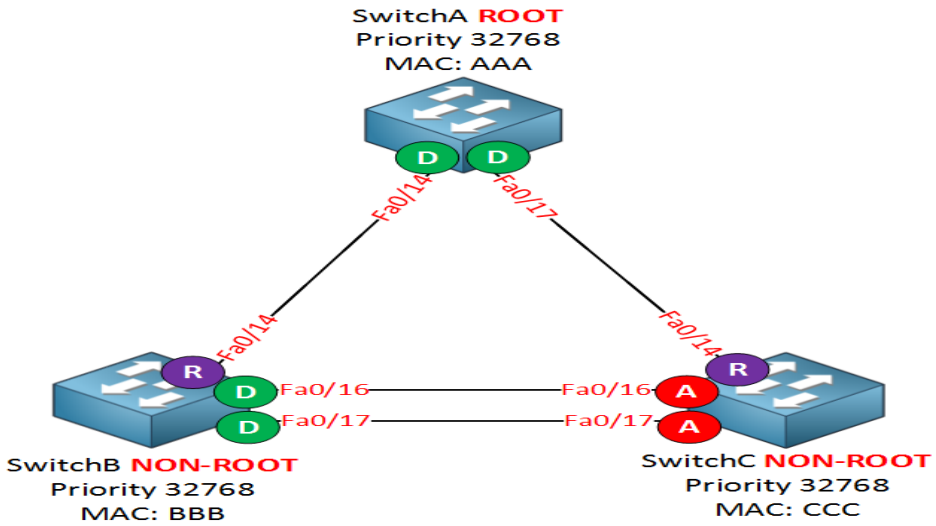
---------- ---- --- --------- -------- --------------------------------

Fa0/14 Root FWD 19 128.14 P2p

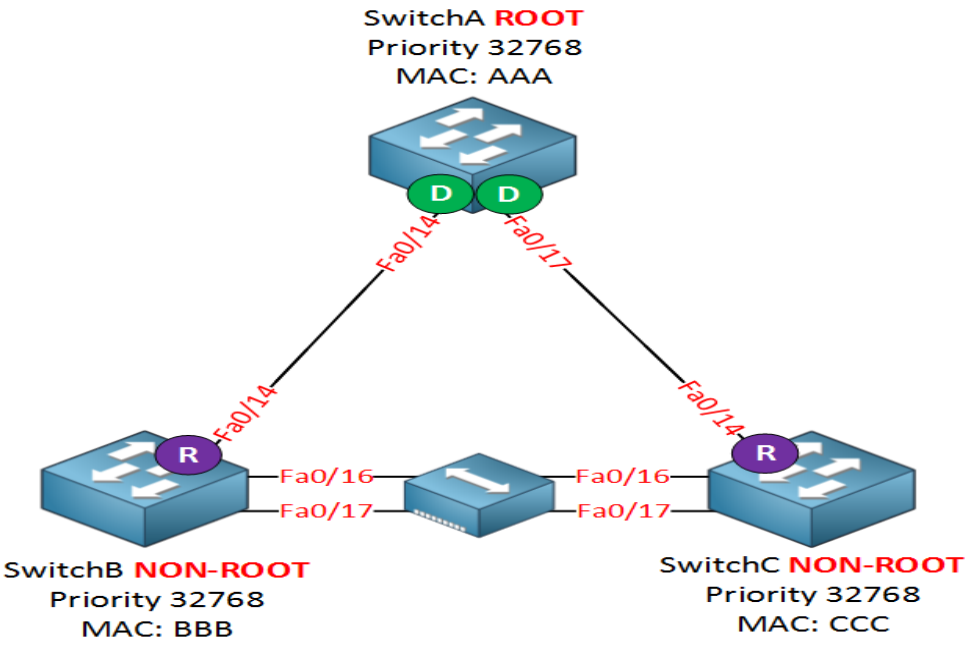
Fa0/16 Altn BLK 19 128.16 P2p

Fa0/17 Altn BLK 19 128.17 P2p

គ្មានអ្វីពិសេសនោះទេ។ យើងទើបតែមាន designated port មួយទៀតនៅលើ SwitchB ហើយមាន Port ផ្លាស់មួយទៀតនៅលើ SwitchC។ សូមបន្ថែម port មួយទៀតមក topology:



ដូច្នេះ topology ដែលប្រើ rapid spanning-tree ដូចទៅនិង classic spanning-tree ដែរ។ ឥឡូវនេះ បង្ហាញពីអ្វីដែលអ្នកមិនធ្លាប់បានដឹងពីមុនមក។ យើងនិងដាក់ Hub មួយនៅចន្លោះ SwitchB និង SwitchC:



សូមពិនិត្យមើលទៅលើ interfaces ម្តងទៀត:

SwitchB#show spanning-tree | begin Interface

Interface Role Sts Cost Prio.Nbr Type

---------- ---- --- --------- -------- --------------------------------

Fa0/14 Root FWD 19 128.5 P2p

Fa0/16 Desg FWD 100 128.3 Shr

Fa0/17 Back BLK 100 128.4 Shr

SwitchC#show spanning-tree | begin Interface

Interface Role Sts Cost Prio.Nbr Type

--------- -------- --------------------------------

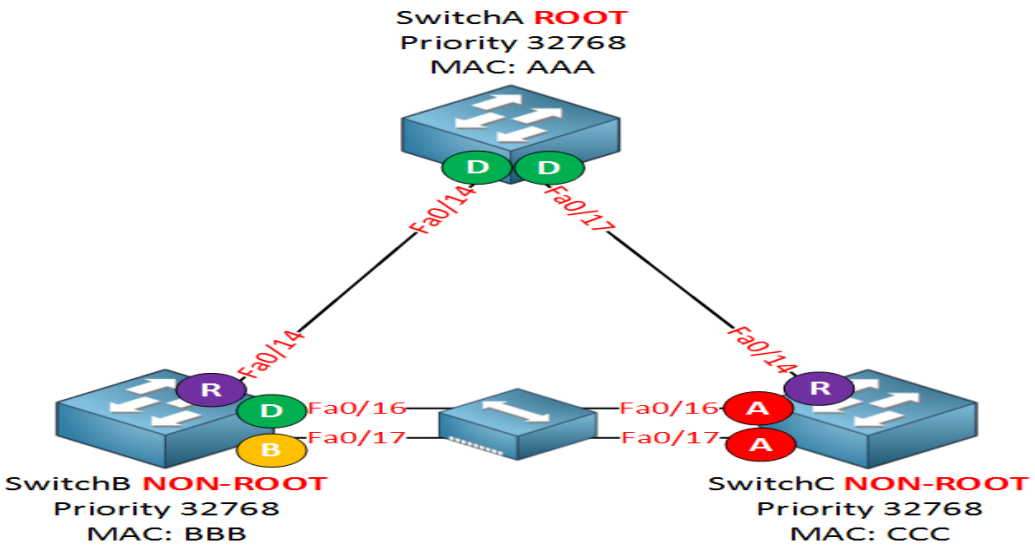
Fa0/14 Root FWD 19 128.5 P2p

Fa0/16 Altn BLK 100 128.3 Shr

Fa0/17 Altn BLK 100 128.4 Shr

នេះគឺជាអ្វីដែលថ្មី។ SwitchB មាន Backup port មួយ។ ដោយសារតែ Hub នៅកណ្តាលនៃ SwitchB និង SwitchC និងដឹងពី BPDUs ដែលវាមានផ្ទាល់ខ្លួន។

អ្នកក៏ដឹងផងដែរថាប្រភេទនៃការភ្ជាប់គឺ shr (shared)។ ពីព្រោះថា Hub ធ្វើឲ្យការប្រាស្រ័យទាក់ទងពី switches មកកាន់ switch តាម Interfaces ជា half duplex។ ខាងក្រោមនេះគឺជា topology:



អ្នកប្រហែលជាមិនបានឃើញថា backup port នៅលើណេតវើកផលិតកម្មពីព្រោះថា hubs វាគឺជាឧបករណ៍មួយគួរឲ្យបារម្ភ។

BPDUsត្រូវបានបញ្ជូនគ្រប់២វិនាទី(hello time) ហើយបើអ្នកចង់ដឹងពីបញ្ហានេះអ្នកអាចប្រើdebug:

SwitchB#debug spanning-tree bpdu

អ្នកក៏អាចប្រើបញ្ជា debug spanning-tree bpdu ដើម្បីមើលពី BPDUs ដែលត្រូវបានបញ្ជូនឬទទួល

SwitchB#

STP: VLAN0001 rx BPDU: config protocol = rstp, packet from FastEthernet0/14, linktype IEEE_SPANNING , enctype 2, encsize 17

STP: enc 01 80 C2 00 00 00 00 11 BB 0B 36 10 00 27 42 42 03

STP: Data

000002023C10010011BB0B360000000000010010011BB0B360080100000140002000F00

STP: VLAN0001 Fa0/14:0000 02 02 3C 10010011BB0B3600 00000000 10010011BB0B3600 8010 0000 1400 0200 0F00

RSTP(1): Fa0/14 repeated msg

RSTP(1): Fa0/14 rcvd info remaining 6

RSTP(1): sending BPDU out Fa0/16

RSTP(1): sending BPDU out Fa0/17

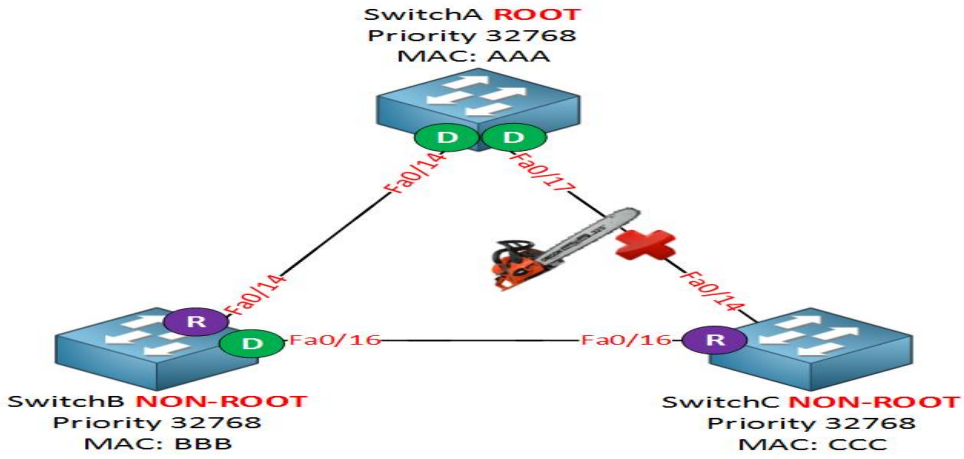
STP: VLAN0001 rx BPDU: config protocol = rstp, packet f

អ្នកក៏អាចឃើញពីអត្ថន័យនៃ BPDU ដូចខាងលើ។ វាមិនសូវសំខាន់នោះទេបើអ្នកចង់ដឹងតែពីអត្ថន័យរបស់ BPDU ប៉ុន្តែវាបង្ហាញអ្នកឲ្យដឹងថា SwitchB កំពុងទទួលនូវ BPDUs ហើយកំពុងតែបញ្ជូនវានៅលើ Interfaces របស់វា។

```
▷ Frame 5 (60 bytes on wire, 60 bytes captured)
▷ IEEE 802.3 Ethernet
▷ Logical-Link Control
▽ Spanning Tree Protocol
    Protocol Identifier: Spanning Tree Protocol (0x0000)
    Protocol Version Identifier: Spanning Tree (0)
    BPDU Type: Configuration (0x00)
  ▷ BPDU flags: 0x00
    Root Identifier: 32769 / 00:19:06:ea:b8:80
    Root Path Cost: 0
    Bridge Identifier: 32769 / 00:19:06:ea:b8:80
    Port identifier: 0x8001
    Message Age: 0
    Max Age: 20
    Hello Time: 2
    Forward Delay: 15
```

បើអ្នកចង់ពិនិត្យពីអត្ថន័យរបស់ BPDU មួយ គេបានសំណូមពរឲ្យប្រើ wireshark ។ វាបង្ហាញពីអ្វីៗដែលមាននៅក្នុងរចនាសម្ព័ន្ធ ។ អ្នកមិនចាំបាច់ចាប់យកនូវ BPDU មួយនោះទេ បើអ្នកមិនចូលចិត្តវា ។ wireshark website មានកត់ត្រាពី packet captures ជាច្រើន ។

យើងនិងបង្ហាញនូវការផ្តាច់ការភ្ជាប់រវាង SwitchA និង SwitchC ដើម្បីបង្ហាញពី rapid spanning tree ដំណើរការ:



SwitchA(config)#interface fa0/17

SwitchA(config-if)#shutdown

ជាដំបូងបិទ (Shutdown) ចំពោះ fa0/17 interface នៅលើ SwitchA

SwitchC#

RSTP(1): updt rolesroot port Fa0/14 is going down

RSTP(1): Fa0/16 is now root port

SwitchC បានដឹងថាមានអ្វីមួយខុសកើតឡើងជាមួយ root port ភ្លាមហើយវានិងប្តូរ fa0/16 interface ពី alternate port មកជា root port។ វាសមមូលទៅនិង UplinkFast ចំពោះ classic spanning tree ប៉ុន្តែវាត្រូវបានបើកតាមលំនាំដើមសម្រាប់ rapid spanning tree។

Spanning-Tree RootGuard

RootGuard ធ្វើឲ្យវាមិនអាចទទួលបាននូវ root bridge ថ្មីមួយទៀត។ BPDU ត្រូវបានបញ្ជូននិងដំណើរការជាធម្មតា ប៉ុន្តែបើ Switch មួយបញ្ជូននូវ BPDU មួយភ្លាមជាមួយ Bridge ID ដែលជាអត្ថិភាព វាមិនទទួលយកជា root bridge នោះទេ។ តាមធម្មតា SW2 អាចក្លាយជា root bridge ដោយសារតែវាមាន bridge ID ល្អប្រសើរជាងគេ។ ជាសំណាងល្អយើងមាន RootGuard នៅលើ SW3 ដូច្នេះវាមិនអាចកើតមានឡើងនោះទេ។ សូមពិនិត្យទៅលើ topology ខាងក្រោម:

បង្ហាញពីរបៀបនៃការដោយប្រើ SW2 និង SW3 ។ ជាដំបូងយើងត្រូវប្រាកដថា SW3 មិនមែនជា root bridge ។

SW2(config)#**spanning-tree vlan 1 priority 4096**

Now we'll enable rootguard on SW2:

SW2(config)#**interface fa0/16**

SW2(config-if)#**spanning-tree guard root**

%SPANTREE-2-ROOTGUARD_CONFIG_CHANGE: Root guard enabled on port FastEthernet0/16.

យើងទទួលបាននូវសារថាវាត្រូវបានបើក។ សូមបើក debug ។ ដូច្នេះយើងអាចឃើញអ្វីដែលកើតមានឡើង។

SW2#**debug spanning-tree events**

Spanning Tree event debugging is on

៩-៤-២-EtherChannel

EtherChannel គឺជាបច្ចេកវិទ្យានៃការបញ្ចូល Physical ports របស់ Switch, Routers និង Servers ពីរឬច្រើនបង្កើតបានជា Port តែមួយ។ វាផ្តល់ឲ្យនូវ Link ដែលមានកម្រិតខ្ពស់ហើយមាន Fault-tolerant។ មាន

Protocols ពីរដែលត្រូវប្រើសម្រាប់ចរចារគឺ EtherChannel និង Link Aggregation។ យើងអាច Configure ចំពោះ EtherChannel តាមបីរបៀបទៅលើ Cisco Switch ។

១-ប្រើ Port Aggregation Protocol (PAgP) ដែលជាកម្មសិទ្ធិរបស់ Cisco

២-IEEE Link Aggregation Control Protocol (LACP) ដែលជាស្តង់ដា Protocol

៣-ការ Configure ចំពោះ EtherChannel ដោយអ្នកគ្រប់គ្រង(មិនប្រើ Protocol ខាងលើ)

គេមាន Switches ពីរដែលភ្ជាប់គ្នាទៅវិញទៅមកតាម Gigabit Ethernet ចាប់ពី Port 1 ដល់ Port លេខ២ ។

ដើម្បី Configure ទៅលើ EtherChannel ដោយប្រើ Port Aggregation Protocol (PAgP) នៅក្នុង Cisco Switch គេអនុវត្តដូចខាងក្រោម:

Switch omnisecu.com.SW1

omnisecu.com.SW1>

omnisecu.com.SW1>**enable**

omnisecu.com.SW1#**configure terminal**

Enter configuration commands, one per line. End with CNTL/Z.

omnisecu.com.SW1(config)#interface range gigabitEthernet 0/1 - 2

omnisecu.com.SW1(config-if-range)#**channel-group 1 mode desirable**

omnisecu.com.SW1(config-if-range)#**channel-protocol pagp**

omnisecu.com.SW1(config-if-range)#**exit**

omnisecu.com.SW1(config)#exit

omnisecu.com.SW1#

Switch omnisecu.com.SW2

omnisecu.com.SW2>

omnisecu.com.SW2>**enable**

omnisecu.com.SW2#**configure terminal**

Enter configuration commands, one per line. End with CNTL/Z.

omnisecu.com.SW2(config)#**interface range gigabitEthernet 0/1 - 2**

omnisecu.com.SW2(config-if-range)#**channel-group 1 mode desirable**

omnisecu.com.SW2(config-if-range)#**channel-protocol pagp**

omnisecu.com.SW2(config-if-range)#**exit**

omnisecu.com.SW2(config)#**exit**

omnisecu.com.SW2#

omnisecu.com.SW1(config-if-range)#channel-group 1 mode desirable specifies the EtherChannel Group Number as 1 and the Port Aggregation Protocol (PAgP) Channel mode as Desirable

Run "show ip interface brief" from Global Configuration mode to find the new EtherChannel virtual interface, Port-channel 1 as shown below.

omnisecu.com.SW2#show ip interface brief

Interface IP-Address OK? Method Status Protocol

<output_omitted>

GigabitEthernet0/1 unassigned YES unset up up

GigabitEthernet0/2 unassigned YES unset up up

Vlan1 unassigned YES unset administratively down down

Port-channel 1 unassigned YES unset up up

omnisecu.com.SW2#

ជំពូកទី១០

Routing

១០-១-សេចក្តីផ្តើមចំពោះ Routers និង Routing

នៅក្នុងមេរៀននេះយើងនិងពិនិត្យមើលពីភាពខុសគ្នារវាង Switches នឹង routers ហើយនិងពន្យល់ពីមូលដ្ឋានគ្រឹះនៃ Routing ។

ជាដំបូង តើ Router ជាអ្វីឬ តើអ្វីជា Routing?

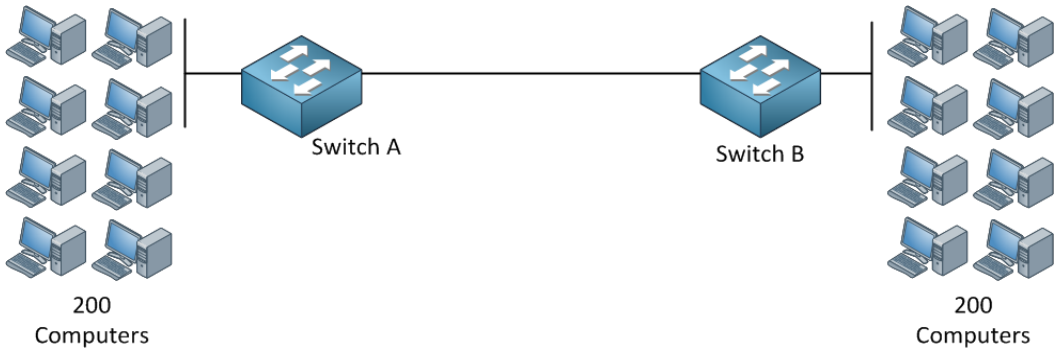
ពាក្យថា Switch “switches” ហើយ Router “route” ប៉ុន្តែវាមានន័យដូចម្តេច?

យើងបានឃើញ Switches ហើយបានសិក្សាពីវាគឺវាអាច “switch” បានដោយពឹងផ្អែកទៅលើ MAC address ។ ចំណាប់អារម្មណ៍ចំពោះ Switch គឺត្រូវដឹងនៅពេលដែល Ethernet frame មួយបានចូលទៅក្នុង Interface របស់វាដែលវាបញ្ជូន Ethernet frame នេះដោយពិនិត្យទៅលើ destination MAC address។ Switches ធ្វើការសម្រេចចិត្តដោយពឹងផ្អែកទៅលើ Data Link layer information (layer 2) ។

Routers មានកិច្ចការប្រហាក់ប្រហែលគ្នាដែរ ប៉ុន្តែនៅពេលនេះយើងពិនិត្យទៅលើ IP packets ហើយអ្នកក៏បានហៅវាថាជា IP ដែលស្ថិតនៅលើ network layer (layer 3)។ Router ពិនិត្យទៅលើ IP address របស់គោលដៅដែលមាននៅក្នុង IP packet ហើយបញ្ជូនចេញតាម interface ដ៏ត្រឹមត្រូវ។

អ្នកប្រហែលគិតថាតើវាខុសគ្នាត្រង់ណា? ហេតុអ្វីបានជាយើងមិនប្រើ MAC addresses នៅក្រប់ទីកន្លែងទាំងអស់? ហើយអ្វីបានជាយើងត្រូវពិនិត្យទៅលើ IP address?

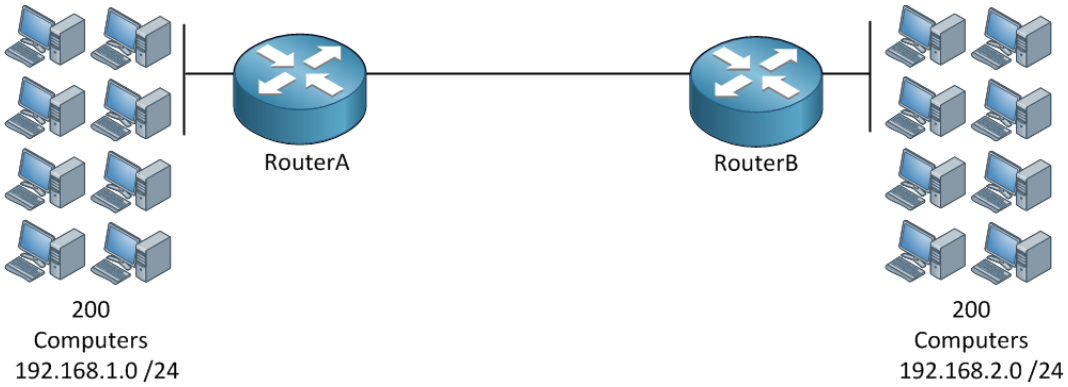
MAC addresses នឹងIP addresses ទាំងពីរនេះគឺមានតែមួយគត់សម្រាប់ឧបករណ៍ ណេតវើក។



តាមរូបភាពខាងលើ យើងមាន Switches ពីរហើយ Switch នីមួយៗមានកុំព្យូទ័រចំនួន ២០ភ្ជាប់ជាមួយវា។ ឥឡូវនេះបើកុំព្យូទ័រចំនួន៤០ ចង់ទាក់ទងជាមួយ Switch នីមួយៗ នោះវាត្រូវដឹងពី 400 MAC addresses។

ចុះចំពោះ Internet វិញតើអ្នកត្រូវគិតថាយ៉ាងណាដែរ?។ វាមានរាប់ពាន់លានឧបករណ៍។ តើវាត្រូវតែបញ្ចូលរាប់ពាន់លាន MAC addresses ទៅក្នុង MAC-address table?ចំពោះឧបករណ៍នីមួយៗនៅក្នុងប្រព័ន្ធ Internet ?

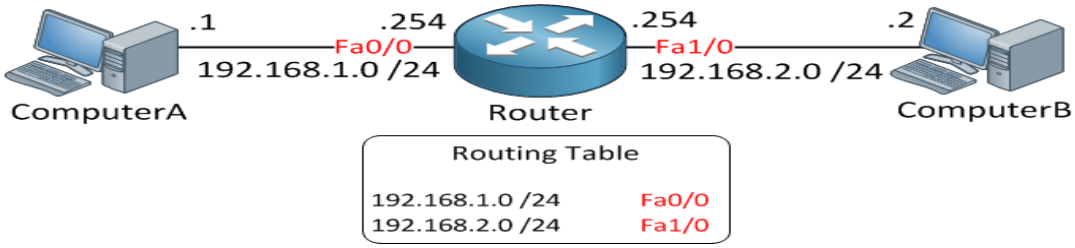
ចំលើយគឺមិនត្រូវការ។ បញ្ហារបស់ Switching គឺវាមិនអាចពង្រីកបាន(Scalable)។ យើងគ្មានការរៀបចំ 48-bit MAC addresses តាមលំដាប់ថ្នាក់នោះទេ។ សូមពិនិត្យមើលទៅលើឧទាហរណ៍ ប៉ុន្តែឥឡូវនេះយើងប្រើ Routers ។



អ្វីដែលយើងមាននៅទីនេះគឺមាន 200 កុំព្យូទ័រនៅខាងឆ្វេងត្រូវបានភ្ជាប់ជាមួយ Router A ហើយមាន 192.168.1.0/24 network។ Router B មាន ២០០កុំព្យូទ័រហើយមាន network address192.168.2.0 /24 ។

Routers “route” ពឹងផ្អែកទៅលើព័ត៌មាន IP ដែលនៅក្នុងឧទាហរណ៍គឺ Router A ដឹងថា ណេតវើក192.168.2.0 /24 នៅក្រោយ Router B។ Router B ត្រូវដឹងថា ណេតវើក192.168.1.0 /24 នៅក្រោយRouter A។

ជំនួសឲ្យការមាន MAC address table មួយដែលមាន 400 MAC addresses យើងត្រូវការតែការបញ្ចូលតែមួយគត់នៅលើ Router នីមួយៗសម្រាប់ ណេតវើកនីមួយៗ។ Switches ប្រើ MAC address tables សម្រាប់បញ្ជូនបន្តចំពោះ Ethernet frames ហើយ Routers ប្រើ routing table មួយដើម្បីដឹងពីទីតាំងដែលត្រូវបញ្ជូនបន្តនូវ IP packets ។ ដរាបណាអ្នកមាន Router ប្រភេទថ្មីមួយ វានិងបង្កើតនូវ routing table មួយ។ ប៉ុន្តែមានតែព័ត៌មានដែលអ្នកត្រូវស្វែងរក Interfaces ដែលភ្ជាប់ដោយផ្ទាល់។ សូមពិនិត្យទៅលើឧទាហរណ៍ដ៏សាមញ្ញាមួយខាងក្រោម:



ខាងលើយើងមាន Router មួយនឹងមានកុំព្យូទ័រពីរ:

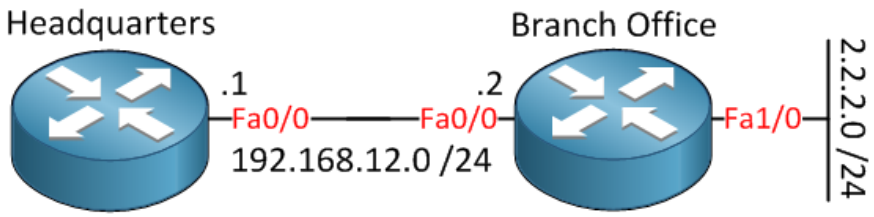
កុំព្យូទ័រ A មាន IP address 192.168.1.1 ហើយត្រូវបាន Configure ជាមួយ IP address 192.168.1.254 ជា default gateway

កុំព្យូទ័រB មាន IP address 192.168.2.2 ហើយត្រូវបាន IP address 192.168.2.254 ជា default gateway។ នៅលើ Router របស់យើង យើងបាន Configure IP address 192.168.1.254 នៅលើ interface FastEthernet 0/0 ហើយមាន IP address 192.168.2.254 នៅលើ interface FastEthernet 1/0។ ដោយសារតែយើងបាន Configure នូវ subnet mask ដែលមាន IP addresses ដែល router ដឹងពីnetwork addresses នឹងក្បាទុក Addresses ទាំងនេះនៅក្នុង routing table របស់វា។

១0-១-១-របៀប configure static route នៅលើ Cisco IOS Router

នៅក្នុងមេរៀននេះយើងពិនិត្យមើលចំពោះ static routes ហើយត្រូវ Configure វា។

សូមពិនិត្យមើលទៅលើ topology ខាងក្រោម:



សូមពិនិត្យមើលណេតវើកនៅក្នុងរូបភាពខាងលើ។ យើងមានណេតវើកមួយដែលមាន Sites ចំនួនពីរគឺការិយាល័យកណ្តាលនិងការិយាល័យសាខាមួយ។

ការិយាល័យកណ្តាលត្រូវបានភ្ជាប់ជាមួយការិយាល័យសាខា។ នៅក្រោយការិយាល័យសាខាគឺជាណេតវើកមួយដែលមាន network address2.2.2.0 /24។ យើងចង់ធ្វើឲ្យដឹងច្បាស់ថាការិយាល័យកណ្តាលអាចភ្ជាប់មកកាន់ណេតវើក 2.2.2.0/24 ។

សូមពិនិត្យមើលពីរបៀប Configure ណេតវើកនេះដោយប្រើ static route:

```
Headquarters>enable

Headquarters#configure terminal
```

ជាដំបូងយើងត្រូវចូលទៅក្នុង configuration mode

```
Headquarters(config)#interface FastEthernet 0/0

Headquarters(config-if)#no shutdown

Headquarters(config-if)#ip address 192.168.12.1 255.255.255.0

Branch>enable

Branch#configure terminal

Branch(config)#interface fastEthernet0/0
```

Branch(config-if)#no shutdown

Branch(config-if)#ip address 192.168.12.2 255.255.255.0

Branch(config-if)#exit

Branch(config)#interface fastEthernet 1/0

Branch(config-if)#no shutdown

Branch(config-if)#ip address 2.2.2.2 255.255.255.0

បន្ទាប់មកយើងនិង Configure IP addresses នៅលើ Interfaces។ សូមកុំភ្លេចប្រើបញ្ជា “no shutdown” នៅលើ interfaces។ សូមពិនិត្យ routing tables របស់ routers ទាំងពីរ។

Headquarters#show ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1,

ia - IS-IS inter area, * - candidate default,

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C 192.168.12.0/24 is directly connected, FastEthernet0/0

Branch#show ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1,

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C 192.168.12.0/24 is directly connected, FastEthernet0/0

2.0.0.0/24 is subnetted, 1 subnets

C 2.2.2.0 is directly connected, FastEthernet1/0

ការប្រើបញ្ជា show ip route សម្រាប់មើលពី routing table។ នេះគឺជាអ្វីដែល Router ប្រើសម្រាប់សម្រេចចិត្តត្រូវ បញ្ជូនបន្ត IP packets ទៅដល់កន្លែងនោះ។ តាមលំនាំដើម router មួយដឹងពី Netwok ដែលមានភ្ជាប់ដោយផ្ទាល់ ប៉ុណ្ណោះ។ យើង configure នូវ IP address និង subnet mask នៅលើ Interface ។ ដូច្នេះ router ក៏ស្គាល់ពី network address។

Router នៅការិយាល័យកណ្តាលស្គាល់អំពី ណេតវើក192.168.12.0/24

Router នៅការិយាល័យសាខា ណេតវើក192.168.12.0/24 and 2.2.2.0/24

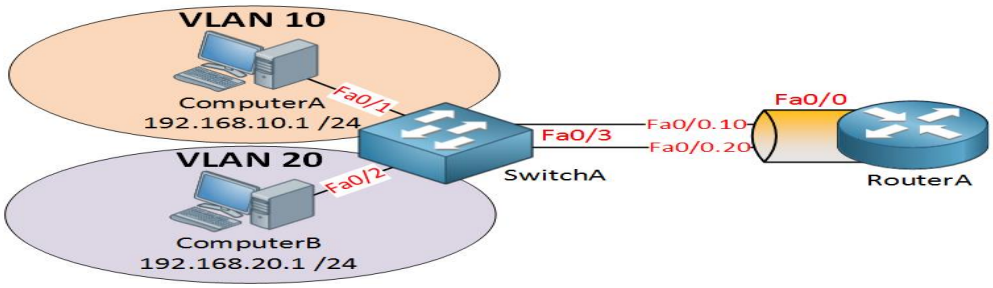
នៅពេលនេះការិយាល័យកណ្តាលរបស់យើងមាន Router ដែលមិនអាចភ្ជាប់មកកាន់ណេតវើក 2.2.2.0/24 ពីព្រោះថាគ្មានអ្វីបញ្ចូលនៅក្នុង routing table។

១០-២-InterVLAN Routing

នៅក្នុងមរៀននេះយើងនិងពិនិត្យទៅលើ routing រវាង VLANs។ នៅពេលយើងចង់ឲ្យវាទាក់ទងរវាង VLANs ផ្សេងគ្នា យើងនិងត្រូវការនូវឧបករណ៍ដែលអាច route បាន។ យើងអាចប្រើ Router ខាងក្រៅមួយ។ ប៉ុន្តែ វាអាចប្រើ Switch ដែលមាន Layers ច្រើន(multilayer switch (aka layer 3 switches)។

សូមពិនិត្យមើលទៅលើ Options ផ្សេងគ្នា។

Router on a Stick



នេះគឺជាការ Setup router on a stick ។ SwitchA មាន VLANs ពីរ។ ដូច្នេះយើងមាន Subnets ពីរ។ បើយើង ចង់ឲ្យវាទាំងពីរទាក់ទងគ្នា យើងត្រូវតែមានឧបករណ៍មួយដែលអាច Route បាន។ នៅក្នុងឧទាហរណ៍ យើងប្រើ Router មួយ។ Router A និងត្រូវការប្រើសម្រាប់ VLANs ទាំងពីរ។ ដូច្នេះយើងអាចបង្កើតនូវ 802.1Q trunk រវាង SwitchA និង RouterA។ នេះគឺជារបៀប configure វា:

SwitchA(config)#interface fa0/3

SwitchA(config-if)#switchport trunk encapsulation dot1q

SwitchA(config-if)#switchport mode trunk

SwitchA(config-if)#switchport trunk allowed vlan 10,20

នេះបង្ហាញពីរបៀបនៃការ Configure SwitchA ។ យក Interface fa0/3 ជា trunk port និងសម្រាប់វិធានការសុវត្ថិភាព ។ ដូច្នេះមានតែ VLAN 10 និង 20 ត្រូវបានអនុញ្ញាត ។

RouterA(config)#interface fa0/0.10

RouterA(config-subif)#encapsulation dot1Q 10

RouterA(config-subif)#ip address 192.168.10.254 255.255.255.0

RouterA(config)#interface fa0/0.20

RouterA(config-subif)#encapsulation dot1Q 20

RouterA(config-subif)#ip address 192.168.20.254 255.255.255.0

បង្កើតនូវ sub-interfaces ពីរនៅលើ router មួយនិងប្រាប់វាឲ្យដឹងថា VLAN មួយណាដែលវាស្ថិតនៅ ។សូមកុំភ្លេចបន្ថែម IP address សម្រាប់ VLAN នីមួយៗ ។

RouterA#show ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static route

o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C 192.168.10.0/24 is directly connected, FastEthernet0/0.10

C 192.168.20.0/24 is directly connected, FastEthernet0/0.20

The router will be able to route because these two networks are directly connected.

C:\Documents and Settings\computerA>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix . :

IP Address. . . . . : 192.168.10.1

Subnet Mask . . . . . : 255.255.255.0

Default Gateway . . . . . : 192.168.10.254

C:\Documents and Settings\computerB>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix . :

IP Address. . . . . : 192.168.20.1

Subnet Mask . . . . . : 255.255.255.0

Default Gateway . . . . . : 192.168.20.254

Don't forget to set your IP address and gateway on the computers.

Let's try a ping:

C:\Documents and Settings\computerA>ping 192.168.20.1

Pinging 192.168.20.1 with 32 bytes of data:

Reply from 192.168.20.1: bytes=32 time<1ms TTL=128

Reply from 192.168.20.1: bytes=32 time<1ms TTL=128

Reply from 192.168.20.1: bytes=32 time<1ms TTL=128

Reply from 192.168.20.1: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

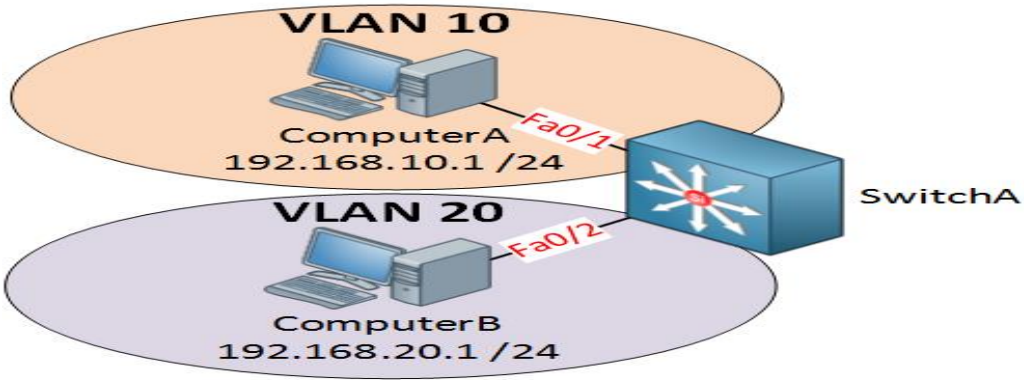
Minimum = 0ms, Maximum = 0ms, Average = 0ms

នេះបានបង្ហាញពីអ្វីដែលអ្នកបានធ្វើវា។ ដូច្នេះហេតុអ្វីបានជាអ្នកត្រូវការនូវដំណោះស្រាយបែបនេះ?

វាអស់តម្លៃថោក។ អ្នកមិនត្រូវការនូវ Switch ដែលមាន Layer ច្រើននោះទេសម្រាប់ Routing។ ត្រូវការត្រឹម Switch ដែលមាន 2 layers អាចធ្វើបាន។ Cisco Catalyst 2960 គឺជា layer 2 switch ចំណែកឯ Switch ដែលមាន Layer ច្រើនហើយមានតម្លៃថោកគឺ Cisco Catalyst 3560។

គុណវិបត្តិនៃដំណោះស្រាយនេះគឺថា router របស់អ្នកគឺជាចំណុចបរាជ័យតែមួយគត់ហើយចរាចរហូរចុះឡើងនៅលើបណ្តាញតែមួយដែលអាចបណ្តាលឲ្យមានការកកស្ទះ(congestion)។ នេះគឺជាការ Configure នៃឧបករណ៍នីមួយៗ។ តើយើងអាចមានដំណោះស្រាយផ្សេងទៀតឬទេ?

SVI (Switch Virtual Interface)



នេះគឺជារូបភាពនៃ Switch ដែលមាន Layer ច្រើន។ Switch នេះមានសមត្ថភាពក្នុងការ Routing បាន។ យើងបាន Configure អ្វីដែលហៅថា SVI (Switch Virtual Interface) សម្រាប់ VLAN នីមួយៗនិងដាក់ IP address ឲ្យវា។ IP address នេះអាចត្រូវបានប្រើសម្រាប់កុំព្យូទ័រដែលប្រើជា default gateway។

នេះគឺជារបៀបនៃការ Configure វា:

SwitchA(config)#ip routing

SwitchA(config)#interface vlan 10

SwitchA(config-if)#no shutdown

SwitchA(config-if)#ip address 192.168.10.254 255.255.255.0

SwitchA(config)#interface vlan 20

SwitchA(config-if)#no shutdown

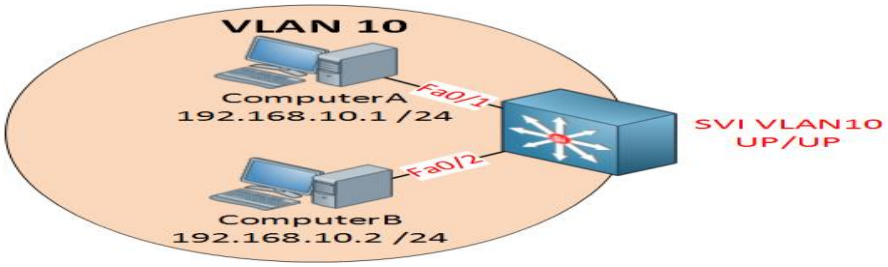
SwitchA(config-if)#ip address 192.168.20.254 255.255.255.0

ចាប់ផ្តើមបើក Routing ដោយប្រើបញ្ជា ip routing ។ បើអ្នកភ្លេចវា Switch និងមិនបង្កើតបាននូវ routing table នោះទេ។ ជំហានបន្ទាប់គឺត្រូវបង្កើតនូវ SVI សម្រាប់ VLAN 10 នឹង 20 ហើយ Configure IP addresses ឲ្យវា។ ការ Configure នេះអាចដូចគ្នា បើអ្នកធ្លាប់ធ្វើការជាមួយ Switch ដែលមានពីរ Layer ពីមុន។ នៅលើ Switch ដែលមានពីរ Layer ដូចជា Cisco Catalyst 2950/2960 ។ យើងក៏មាន SVI ។ ប៉ុន្តែអ្នកអាចប្រើវាសម្រាប់ការគ្រប់គ្រងពីចម្ងាយ។

នៅពេលអ្នកបង្កើតនូវ SVI ហើយប្រើបញ្ជា no shutdown តាមធម្មតាវានិង “up” ពីព្រោះថាវា virtual interface ប៉ុណ្ណោះ។ មានតម្រូវការមួយចំនួនឬវានិងបង្ហាញថា “down”:

VLAN ត្រូវតែមានជាស្រេចនៅក្នុង VLAN database ហើយវាគួរតែ active

យ៉ាងហោចណាស់មាន Access ឬ trunk port មួយដែលប្រើ VLAN នេះឲ្យមានសកម្មភាពហើយវាស្ថិតនៅក្នុង spanning-tree forwarding mode។ និយាយឲ្យខ្លី VLAN ត្រូវតែ Active ឬ SVI និង Down ។

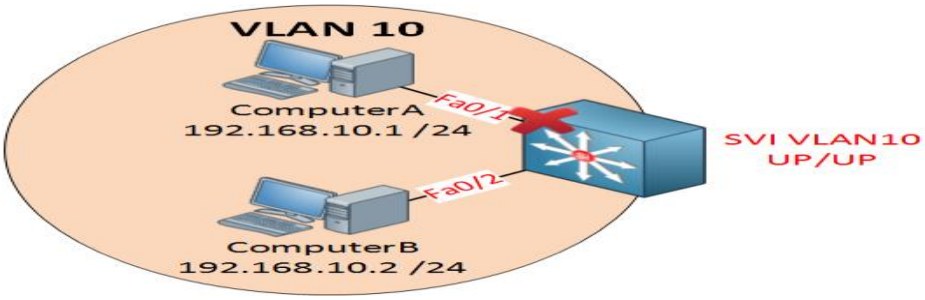


យើងមានកុំព្យូទ័រពីរនៅក្នុង VLAN 10 ហើយបង្កើត SVI មួយសម្រាប់ VLAN 10 ។

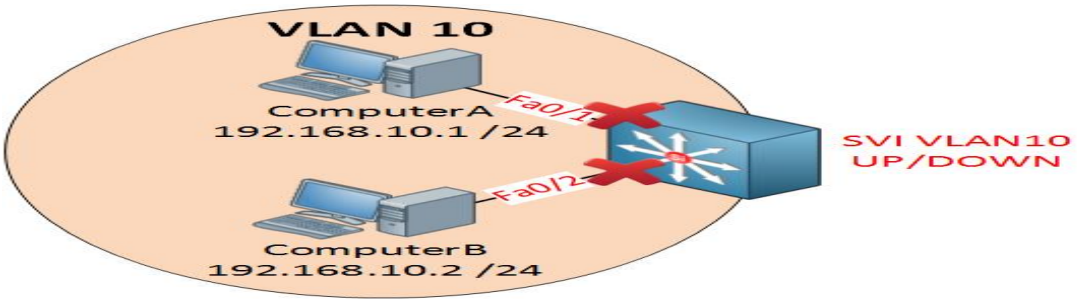
SwitchA#show ip interface brief vlan 10

Interface	IP-Address	OK?	Method	Status	Protocol
Vlan10	192.168.10.254	YES	manual	up	up

អ្នកនិងឃើញពីលក្ខណៈរបស់គឺ up/up ដែលល្អ។



បើយើង shutdown ចំពោះ interface មួយក្លាម នោះវានិងគ្មានអ្វីត្រូវផ្លាស់ប្តូរនោះទេ។ SVI និងបង្ហាញឲ្យឃើញថា up/up ពីព្រោះ interface fa0/2 នឹង active។



SwitchA#show ip interface brief vlan 10

Interface IP-Address OK? Method Status Protocol

Vlan10 192.168.10.254 YES manual up down

នៅពេលយើង Shutdown interfaces ទាំងពីរក្លាម។ គ្មានអ្វីត្រូវ Active បន្តទៀតទេនៅក្នុង VLAN 10។ ជាលទ្ធផលគឺ SVI នឹងក្លាយជា up/down។

ឥឡូវនេះយើងចង់ដក Interface មួយចេញពី SVI state។ ស្រមៃថា យើងចង់ដឹងថាតើមានអ្វីកើតឡើងចំពោះ interface fa0/2 ដែលគ្មានឥទ្ធិពលទៅលើ SVI state:

SwitchA(config)#interface fa0/2

SwitchA(config-if)#switchport autostate exclude

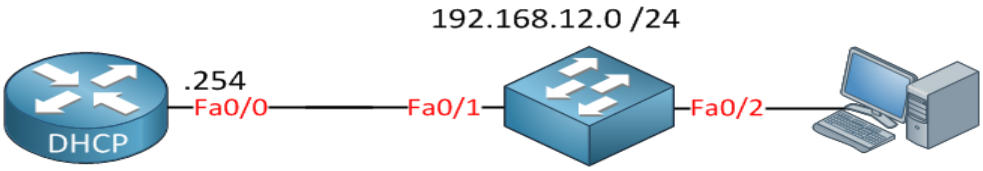
យើងអាចប្រើបញ្ជា switchport autostate exclude ។ នេះមានន័យថាវាគ្មានឥទ្ធិពលទៅលើលក្ខណៈនៃ SVI interface បន្តទៀតទេ។ Fa0/1គឺជា Interface តែមួយគត់ដែលអាចមានឥទ្ធិពលទៅលើលក្ខណៈនៃ SVI ដរាបណាវា down ។ អ្នកនឹងឃើញថាលក្ខណៈរបស់ SVIdown ផងដែរ។ទោះបីជាfa0/2នៅតែupនិងដំណើរការ។

ទោះបីជា SVI គ្រប់គ្រាន់ក៏ដោយ មានវិធីសាស្ត្រមួយទៀតដែលយើងអាចប្រើ Switch ដែលមាន Layer ច្រើនសម្រាប់ Routing ។ តាមលំនាំដើមគ្រប់ Interfaces ទាំងអស់នៅលើ Switch មួយគឺជា switchports (layer 2)។ ប៉ុន្តែយើងអាចប្តូរវាមកជា routed ports (layer 3)បាន។ routed port មួយគឺជា Interface ដូចគ្នាទៅនិងអ្វីដែលយើងបានប្រើនៅលើ Router មួយ។

១0-៣-របៀប configure DHCP Server នៅលើ Cisco IOS

Cisco IOS routers នឹង layer 3 switches គេអាច Configure វាជាDHCP server។ វាងាយណាស់ដើម្បី Configure វា។

ពិនិត្យមើល Topology ខាងក្រោមនេះ



ខាងលើយើងមាន Router មួយដែលប្រើជា DHCP ។ Router និងកុំព្យូទ័រត្រូវបានភ្ជាប់ជាមួយគ្នាដោយប្រើ Switch មួយហើយស្ថិតក្នុង VLAN ជាមួយគ្នា។ យើងប្រើ 192.168.12.0 /24 subnet សម្រាប់ការបង្ហាញនេះ។

DHCP(config)#interface fastEthernet 0/0

DHCP(config-if)#no shutdown

DHCP(config-if)#ip address 192.168.12.1 255.255.255.0

Now let's configure DHCP server:

DHCP(config)#ip dhcp pool MYPOOL

DHCP(dhcp-config)#ណោតវើក192.168.12.0 255.255.255.0

គេប្រើបញ្ជា ip dhcp pool ដើម្បីបង្កើត DHCP pool មួយហើយដាក់ឈ្មោះឲ្យវា។ DHCP pool នេះនឹងប្រើណោតវើក192.168.12.0 /24។ យើងត្រូវតែធ្វើឲ្យ DHCP server ដំណើរការ។ យើងអាចផ្ទៀងផ្ទាត់ថាយើងមាន DHCP clients ដោយប្រើបញ្ជាដូចខាងក្រោមនេះ:

DHCP#show ip dhcp binding

Bindings from all pools not associated with VRF:

IP address	Client-ID/ Hardware address/ User name	Lease expiration	Type
192.168.12.2	0063.6973.636f.2d63. 6330.372e.3132.3265. 2e30.3030.302d.4661. 302f.30	Mar 02 2002 12:24 AM	Automatic

ដូចអ្នកបានឃើញខាងលើហើយថា យើងមាន DHCP client មួយហើយវាទទួលបាននូវ IP address 192.168.12.2។ នៅក្នុងណោតវើកផលិតកម្ម យើងក៏មាន DHCP ដៃរដើម្បីផ្តល់ព័ត៌មានសំខាន់មួយចំនួនទៅឲ្យ Clients មានដូចជា default gateway, DNS server ។ល។

សូមពិនិត្យមើលពីរបៀបនៃការ Configure:

DHCP(config)#ip dhcp pool MYPOOL

DHCP(dhcp-config)#default-router 192.168.12.1

DHCP(dhcp-config)#dns-server 208.67.222.222

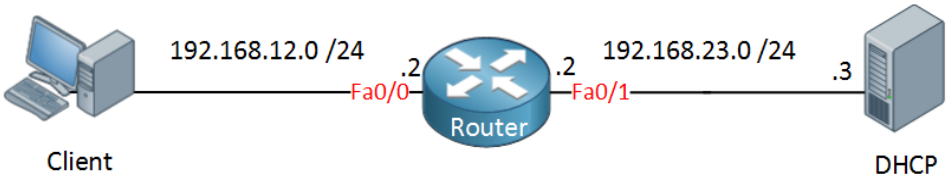
ខាងលើយើងបាន Configure IP address 192.168.12.1 ជា default gateway សម្រាប់ DHCP clients ជាមួយបញ្ជា default-router ។ បញ្ជា dns-server អនុញ្ញាតឲ្យយើងអាចកំណត់ពី DNS server ។

ជាមួយការ Configure នេះ DHCP server និងផ្តល់ IP address .2,3,4,5,6 ។ល។

១0-៣-១-Cisco IOS DHCP Relay Agent

DHCP ត្រូវបានគេប្រើសម្រាប់ផ្តល់ IP addresses ជាស្វ័យប្រវត្តិហើយប្រើនូវ Packets ចំនួន៤ផ្សេងគ្នា។ ដោយសារតែ Host មួយគ្មាន IP address នោះទេ វា broadcast នូវ messages នៅលើ Network ។ បញ្ហាជាមួយ Broadcast គឺថា DHCP server ត្រូវតែស្ថិតនៅក្នុង broadcast domain ជាមួយគ្នា ។ ដោយសារតែ Routers មិនអាចបញ្ជូនបន្តនូវ broadcast packets ។

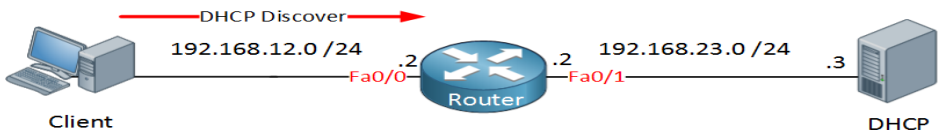
សូមពិនិត្យមើលរូបភាពខាងក្រោម:



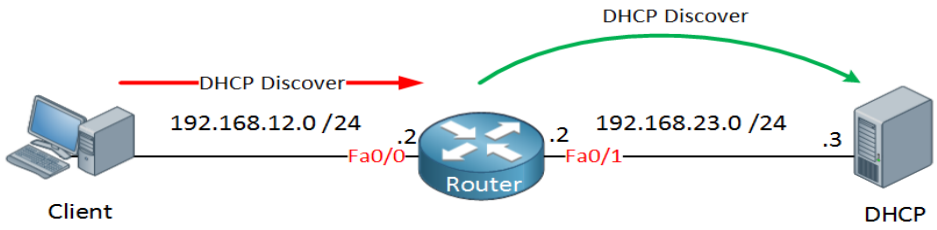
នៅខាងឆ្វេងយើងមាន Client មួយ នៅកណ្តាលមាន Router មួយនិងនៅខាងស្តាំមាន DHCP server មួយ។ Client ដែលចង់ទទួល IP address មួយតាមរយៈនៃ DHCP និងបញ្ជូននូវ broadcast មួយដែលមានឈ្មោះថា DHCP discover message ។ router ក៏ធ្វើកិច្ចការរបស់វាហើយក៏មិនបញ្ជូនបន្តនូវចរាចរណ៍ broadcast នោះទេ ។ ដូច្នេះ DHCP discover និងមិនអាចទាក់ទងជាមួយ DHCP server បានទេ ។

តើយើងដោះស្រាយបញ្ហានេះតាមរបៀបណា ? យើងត្រូវតែមាន DHCP Relay Agent ។ និយាយដោយខ្លី Router នឹងបញ្ជូនបន្តនូវ DHCP requests ពី Clients ទៅឲ្យ DHCP server ។ នៅពេលដែល DHCP server ឆ្លើយតបវិញវានិងបញ្ជូនបន្តនូវ messages នោះត្រឡប់មក Client វិញ ។

សូមតាមដានជំហានដូចខាងក្រោម:

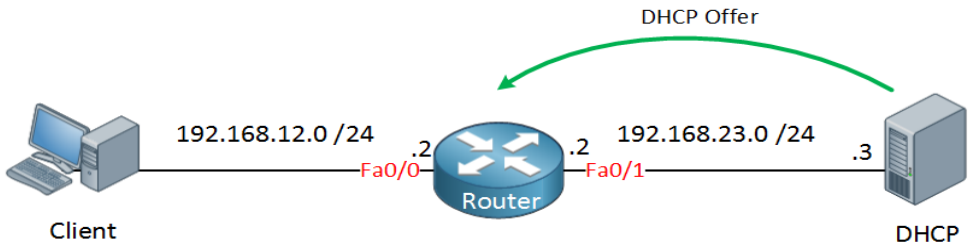


ជាដំបូងគឺ Client និងបញ្ជូននូវ broadcast មួយមានឈ្មោះថា DHCP discover message បន្ទាប់មក Router និងទទួលបាននូវ message ពីព្រោះថាវាស្ថិតនៅក្នុង broadcast domain ជាមួយ client ។ តើមានអ្វីកើតមានឡើងបន្ទាប់ទៀត?

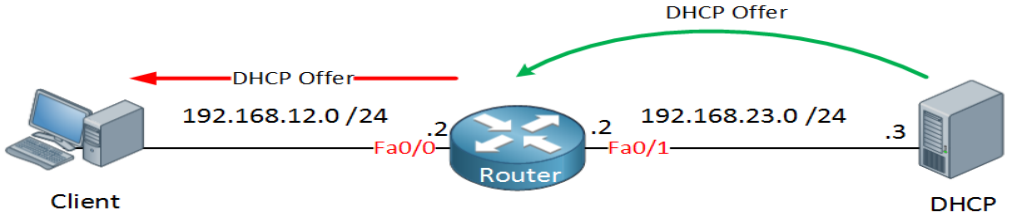


router ក៏ទទួលបាននូវ DHCP discover message នៅលើ FastEthernet 0/0 interface របស់វាហើយវានិងបោះចោលនូវ Packet នេះ។ ជាមួយ DHCP relay agent ត្រូវបានបើក។ វានិងអាចធ្វើកិច្ចការលើសពីនេះទៅទៀត។ វានិងបញ្ជូនបន្តនូវ DHCP discover message ជា unicast packet ហើយក៏ញត់បញ្ចូលផ្នែកមួយហៅថា giaddr (Gateway IP Address) នៅក្នុង DHCP packet។ វានិងស៊កបញ្ចូលនូវ IP address 192.168.12.2 នៅក្នុងផ្នែកនេះពីព្រោះថា វាបានទទួលនូវ DHCP discover នៅលើ FastEthernet 0/0 interface។ ផ្នែក giaddr នេះគឺត្រូវការដោយ DHCP server ឬវាមិនដឹងពី Pool មួយណាដែលត្រូវជ្រើសរើសយក IP address មួយនោះទេ។ ចំពោះ source IP address នៃ unicast packet គឺជា 192.168.12.2 ។

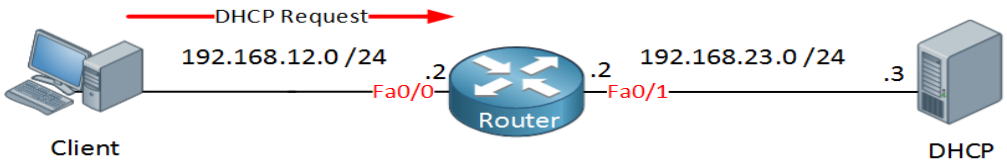
សូមបន្តទៅទៀត:



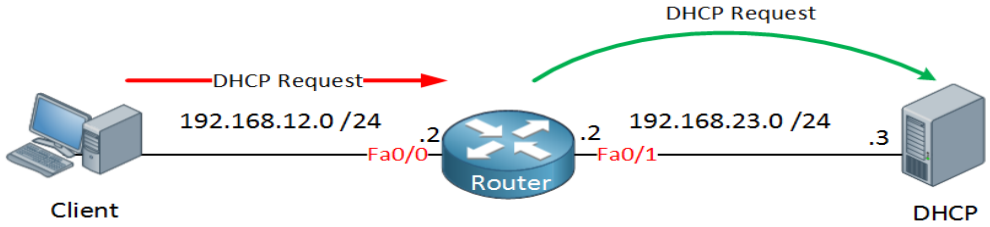
DHCP server បានទទួលនូវ DHCP discover message ហើយក៏បញ្ជូនត្រឡប់មកវិញនូវ DHCP offer message មួយ។ វាគឺជា unicast packet មួយដែលត្រូវបញ្ជូនមក router ។



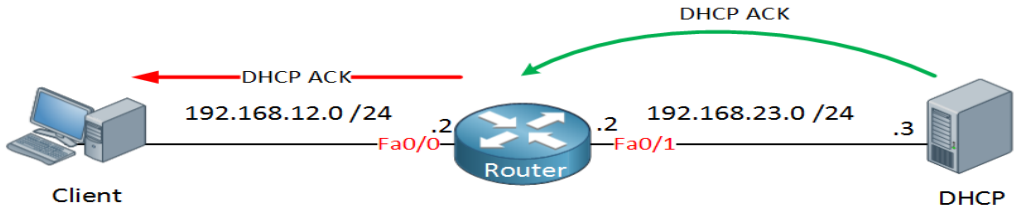
បន្ទាបមក Router ក៏បញ្ជូនបន្តនូវ DHCP offer នៅលើ FastEthernet0/0 interface ជា broadcast មួយ។



client ពេញចិត្តចំពោះអត្ថន័យរបស់ DHCP offer message និងបង្កើតនូវ DHCP request មួយដែលជា broadcast។ router ក៏បានដឹងពី broadcast នេះហើយក៏ធ្វើការ។



វាក៏ដូចទៅនិង DHCP discover message ដែរ។ DHCP request នេះនិងត្រូវបានបញ្ជូនបន្តជា unicast packet មួយ។ ភ្លាមនោះផ្នែក giaddr ក៏ត្រូវបានស៊ីកបញ្ចូលជាមួយ IP address 192.168.12.2។ DHCP server ទទួលនូវ DHCP request ហើយក៏បន្តដំណើរការទៅមុខទៀត។



ជាចុងក្រោយ DHCP server និងបញ្ជូននូវ DHCP ACK មួយឆ្លើយតបទៅនិង DHCP request។ វាត្រូវបានបញ្ជូនទៅឲ្យ Router ដោយប្រើ unicast ហើយ Router និង broadcast វានៅលើ FastEthernet 0/0 interface ។ ដូច្នេះ client ក៏ទទួលយកវា។ ឥឡូវនេះ client ក៏ទទួលបាននូវ IP address ហើយបេសកកម្មក៏ទទួលបានជោគជ័យ។

Configuration

យើងមាន Routers ចំនួន ៣ នៅក្នុង ណេតវើកនេះ។

Client(config)#interface FastEthernet 0/0

Client(config-if)#no shutdown

Router(config)#interface FastEthernet 0/0

Router(config-if)#no shutdown

Router(config-if)#ip address 192.168.12.2 255.255.255.0

Router(config)#interface FastEthernet 0/1

Router(config-if)#no shutdown

Router(config-if)#ip address 192.168.23.2 255.255.255.0

DHCP(config)#interface FastEthernet 0/0

DHCP(config-if)#no shutdown

DHCP(config-if)#ip address 192.168.23.3 255.255.255.0

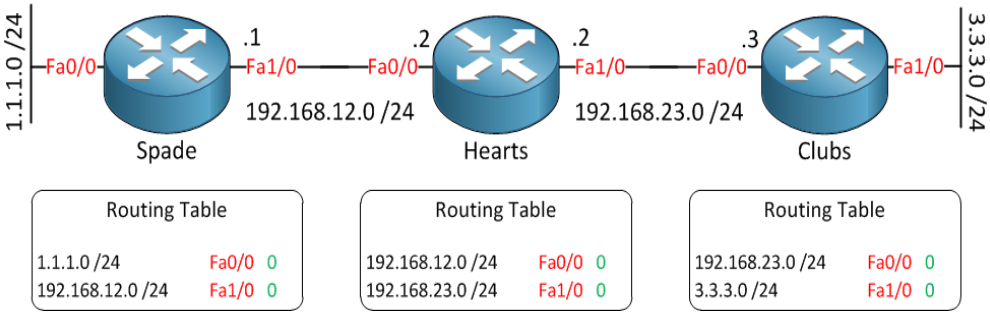
១0-៤-RIP Distance Vector Routing Protocol

RIP គឺជា distance vector routing protocol មួយហើយវាជា routing protocol ដ៏សាមញ្ញមួយដែលត្រូវចាប់ផ្តើមជាមួយវា។ យើងនិងចាប់ផ្តើមដោយយកចិត្តទុកដាក់ទៅលើ distance vector class។ តើ distance vector មានន័យដូចម្តេច ?

Distance គឺជាកំនត់ចម្ងាយដែលនៅក្នុងពិភពពិតនៃ Routing យើងប្រើមេត្រិក

Vector គឺជាទិសដៅដែលនៅក្នុងពិភពនៃ Routing យើងចាត់ទុកថាជា Interface និង IP address របស់ Router បន្ទាប់ដែលត្រូវបញ្ជូនទៅឲ្យ

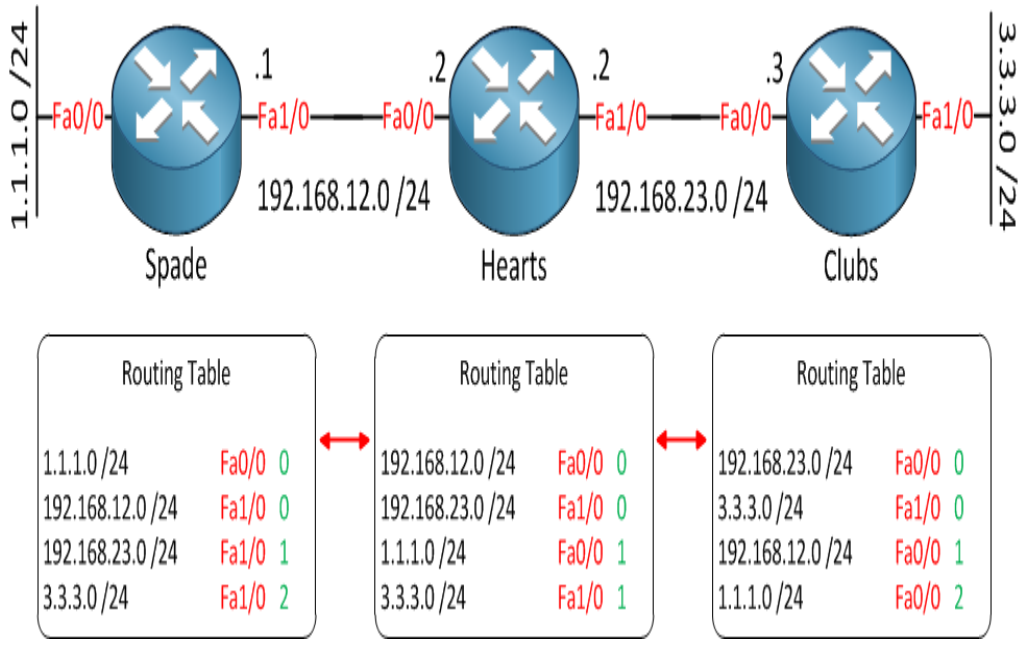
សូមពិនិត្យមើលពីរបៀបដែល distance vector routing protocols ដំណើរការ:



នៅក្នុងរូបភាពយើងមាន Router ចំនួន៣ហើយយើងកំពុងប្រើ distance vector routing protocol (RIP)។ ដូចយើងដំណើរការនៃ Routers វាបង្កើត routing table តាមលំនាំដើម។ ប៉ុន្តែអ្វីដែលវាដឹងនោះគឺ Interfaces ដែលបានភ្ជាប់ដោយផ្ទាល់។

អ្នកបានឃើញថាព័ត៌មាននេះនៅក្នុង Routing table របស់វា។ ពណ៌ក្រហមដែលអ្នកបានឃើញគឺជា Interface ហើយពណ៌បៃតងដែលអ្នកបានឃើញគឺជាមេត្រិក។ RIP ប្រើ Hop count ជាមេត្រិករបស់វា។ វាមានតួនាទីជាអ្នករាប់ចំនួន Routers ដែលវាបានឆ្លងកាត់ដើម្បីទៅដល់គោលដៅ។

យើងនិងប្រើ distance vector routing ។ អ្វីដែលកើតមានឡើងគឺ Routers របស់យើងនិងថតចំលងនូវ routing table របស់វាទៅឲ្យ Routers ជិតខាងដែលភ្ជាប់ដោយផ្ទាល់ ។ ចំពោះ Router ដែលមានឈ្មោះថា Spade និងថតចំលងនូវ Routing table របស់វាទៅឲ្យ Router ដែលមានឈ្មោះថា Hearts ។ Router ដែលមានឈ្មោះថា Hearts និងថតចំលងនូវ Routing table របស់វាទៅឲ្យ Router ដែលមានឈ្មោះថា Clubs និង Routers ផ្សេងៗទៀតដែលនៅជិតៗគ្នានោះ ។ បើ Router មួយទទួលបាននូវព័ត៌មានអំពី ណេតវើកមួយ វាមិនបានដឹងនោះទេ ។ វានិងបន្ថែមព័ត៌មាននេះទៅក្នុង Routing tables ។



សូមពិនិត្យមើលទៅលើ Router ដែលមានឈ្មោះថា Spade ហើយអ្នកនឹងឃើញថាវាមានដឹងពី 192. 168. 23.0/24 និង 3.3.3.0/24 ណេតវើកពី router ដែលមានឈ្មោះថាHearts ។ អ្នកក៏បានដឹងថាវាបានបន្ថែមនូវ interface(Fa1/0)ដើម្បីអាចភ្ជាប់ជាមួយ Networks ទាំងនេះ (ដែលជាផ្នែកនៃ Vector) ហើយអ្នកក៏ឃើញថាវាបានបន្ថែមនូវមេត្រិក (hop count) សម្រាប់ Networks ទាំងនេះ(គឺជាផ្នែកនៃចម្ងាយ)។

ចំពោះ Network192.168.23.0/24គឺមានមួយ Hop ហើយNetwork 3.3.3.0/24 មាន 2 hops ។

អ្នកក៏បានឃើញថា Router ដែលមានឈ្មោះថា Hearts និង router ដែលមានឈ្មោះថា Clubs បានបំពេញចំពោះ Routing tables របស់វា ។ នៅរៀងរាល់ ៣០វិនាទីម្តង Routers របស់យើងនិងបញ្ជូននូវច្បាប់ចំលងទាំងអស់នៃ Routing table ទៅឲ្យ Routers ដែលជាអ្នកជិតខាងរបស់វាដើម្បីធ្វើបច្ចុប្បន្នភាពចំពោះ Routing table របស់វា ។

ដូច្នេះ Routers កំពុងធ្វើការហើយយើងក៏ដឹងពីគោលដៅរបស់វាចំពោះ Networks របស់យើង ។ ចំពោះ Distance vector routing protocols មានភាពខ្សោយចំពោះបញ្ហាមួយចំនួន ។ សូមពិនិត្យមើលទៅលើអ្វីដែលបានកើតមានឡើងដែលជាកំហុស ។

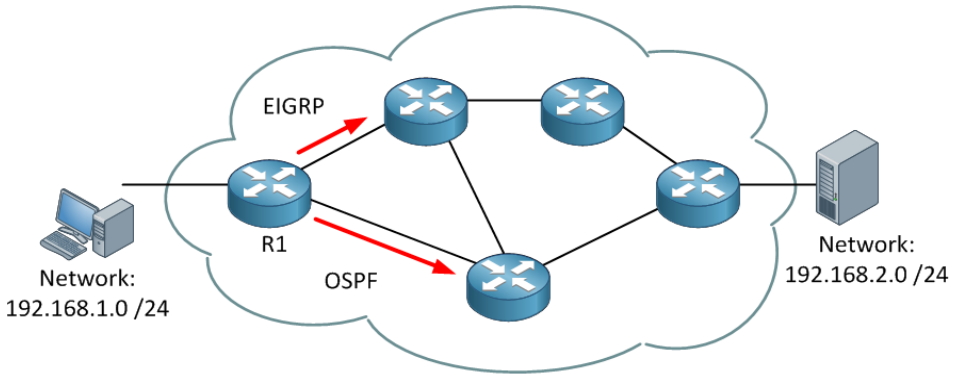
១0-៥-Administrative Distance

Administrative distanceគឺជាជម្រើសមួយនៃសញ្ញាណRoutingដែលអ្នកសិក្សាតែងតែមានការលំបាកយល់ ។

Administrative Distance (AD) គឺជាតម្លៃមួយដែល Routers ប្រើក្នុងគោលបំណងដើម្បីជ្រើសរើសផ្លូវបញ្ជូនដ៏ប្រសើរបំផុតនៅពេលដែលមានផ្លូវបញ្ជូនពីរឬច្រើនខុសគ្នាមកកាន់គោលដៅតែមួយពី Routing Protocols ពីរផ្សេងគ្នា។ AD រាប់ពីទំនុកចិត្តនៃ Routing Protocol មួយ។ AD គឺជាតម្លៃលេខមួយដែលមានតម្លៃចាប់ពី 0 ទៅ ២៥៥ ។ AD តូចបំផុតគឺជាតម្លៃមួយដែល Router បានជ្រើសរើសយក។ ដូច្នេះ តម្លៃនៃ AD ប្រសើរបំផុតគឺ0ហើយតម្លៃដែលមិនល្អគឺ ២៥៥។

Administrative (AD)	Distance	Route Type
0		Connected interface
0 or 1		Static Route
90		Internal EIGRP Route (within the same Autonomous System (AS))
100		IGRP Route
110		OSPF Route
115		IS-IS
120		RIP Route
255		Unknown Route

នៅក្នុងអត្ថបទនេះយើងពន្យល់អ្វីទៅជា Administrative distance និងពីរបៀបដែលវាធ្វើការ។



នៅក្នុងរូបភាពយើងមានណេតវើកមួយដែលកំពុងដំណើរការនូវ Protocols ពីរនៅពេលតែមួយគឺ OSPF និង EIGRP ។ Routing Protocols ទាំងពីរកំពុងផ្តល់ព័ត៌មានទៅឲ្យ R1។ EIGRP ប្រាប់យើងថា Router គួរតែបញ្ជូន IP packets ដែលកំពុងប្រើផ្លូវនៅខាងលើបំផុត។

OSPF ប្រាប់ឲ្យយើងដឹងថា Router គួរតែ IP packets ដែលកំពុងប្រើផ្លូវខាងក្រោម:

តើព័ត៌មាន Routing អ្វីដែលយើងកំពុងប្រើវា? ទាំងពីរ? ប្រើ OSPF ឬ EIGRP?

ចំលើយគឺនៅពេលដែល Routing Protocols ពីរកំពុងផ្តល់ឲ្យយើងនូវព័ត៌មានអំពីគោលដៅណេតវើកដូច គ្នា យើងត្រូវតែធ្វើការសម្រេចចិត្តជ្រើសរើស។ យើងត្រូវពិនិត្យទៅលើ Administrative distance ឬ AD ។

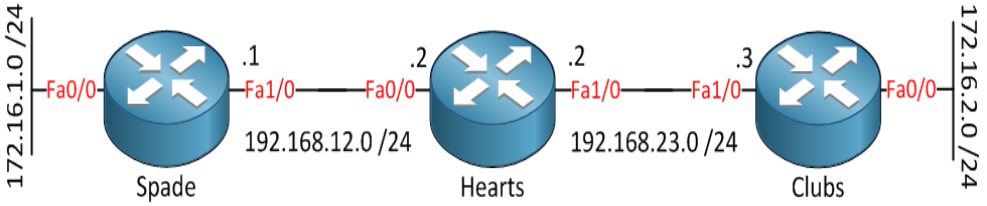
ការប្រៀបធៀបរវាង Classful ជាមួយ Classless Routing Protocols

Routing protocols អាចជា classful ឬ classless:

Classful routing protocols មិនបញ្ជូន subnet mask នៅពេលធ្វើការ updates

Classless routing protocols បញ្ជូន subnet mask ទៅជាមួយនៅពេលធ្វើការ updates

សូមពិនិត្យមើលឧទាហរណ៍



យើងមាន Routers ចំនួន៣ហើយមានបណ្តុំនៃ Networks។ សូមពិនិត្យមើលដោយយកចិត្តទុកដាក់ចំពោះ Networks ដែលយើងមាន:

172.16.1.0 /24

172.16.2.0 /24

192.168.12.0 /24

192.168.23.0 /24

ត្រូវដឹងពី Class A,B និង C

172.16.1.0 នឹង 172.16.2.0 ស្ថិតនៅក្នុង class B

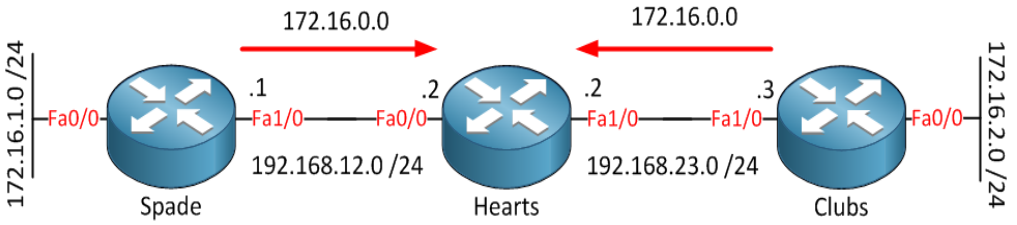
192.168.12.0 នឹង 192.168.23.0 ស្ថិតនៅក្នុង class C

តើ Class B នឹង C មាន subnet mak តាមលំនាំដើមអ្វី?

Class B: 255.255.0.0

Class C: 255.255.255.0

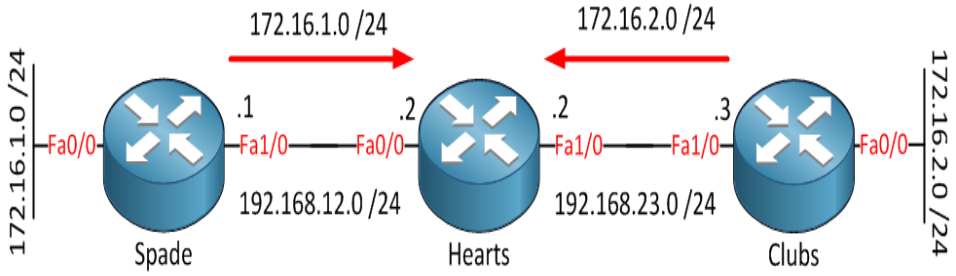
Classful routing protocol នឹងមិនបញ្ជូន Subnet mask ទៅជាមួយការធ្វើបច្ចុប្បន្នភាពនៃ Routing Table នោះទេ។ តើមានអ្វីកើតឡើង?



ចំពោះ Router ដែលមានឈ្មោះថា Spade និង router Clubs មិនបញ្ជូននូវ subnet mask ទៅជាមួយទេនៅពេលធ្វើការ Update ចំពោះ Routing table។ ដូច្នេះវានិងផ្សព្វផ្សាយនូវ classful ណេតវើកដែលជា 172.16.0.0 នៅក្នុងករណីនេះ។ តើមានអ្វីកើតឡើងចំពោះ router Hearts?

វាអាចទាក់ទងជាមួយ172.16.0.0 ណេតវើកដោយបញ្ជូន Packets ទៅខាងឆ្វេងឬខាងស្តាំហើយបើមេត្រិកស្មើគ្នាវានិងព្យាយាមធ្វើឲ្យមានតុល្យភាព។ តាមការពិតវានឹងធ្វើឲ្យមានបញ្ហាកើតឡើង។

Classless routing protocols ផ្សព្វផ្សាយនូវ subnet mask ទៅជាមួយនៅពេល updates:



ដូចដែលអ្នកបានដឹងពី router Spade កំពុងផ្សព្វផ្សាយ 172.16.1.0 subnet ជាមួយ subnet mask។ Router Clubs កំពុងផ្សព្វផ្សាយ 172.16.2.0 subnet ជាមួយ subnet mask ផងដែរ។ តើ routing protocols មួយណាជា classful ឬ classless?

ខាងក្រោយនេះគឺជាការបង្ហាញពីការប្តូរ AD ។

```
Router( config )#router eigrp 12

Router( config-router )#distance eigrp 90 160
```

ខាងលើយើងមាន EIGRP និងបញ្ជា distance ដែលប្រើដើម្បីប្តូរ AD សម្រាប់ EIGRP ។ ចំពោះ EIGRP ខ្នាត់អន្តរជាតិមានតម្លៃ 90 ប៉ុន្តែ EIGRP ខាងក្រៅនិងមានតម្លៃ 160 ។ អ្នកនឹងឃើញពីអ្វីដែលបានប្តូរនៅក្នុង Routing table

```
Router#show ip route eigrp

    3.0.0.0/24 is subnetted, 1 subnets

D EX    3.3.3.0 [160/1734656] via 192.168.12.2, 00:00:30, FastEthernet0/0
```

D EX 192.168.23.0/24 [160/1734656] via 192.168.12.2, 00:00:30,FastEthernet0/0

អ្នកអាចផ្ទៀងផ្ទាត់វាដោយពិនិត្យមើលក្នុង Routing table ។ ណេតវើកខាងក្រៅនៅលើ Router មានតម្លៃ AD ស្មើនឹង១៦០ ។

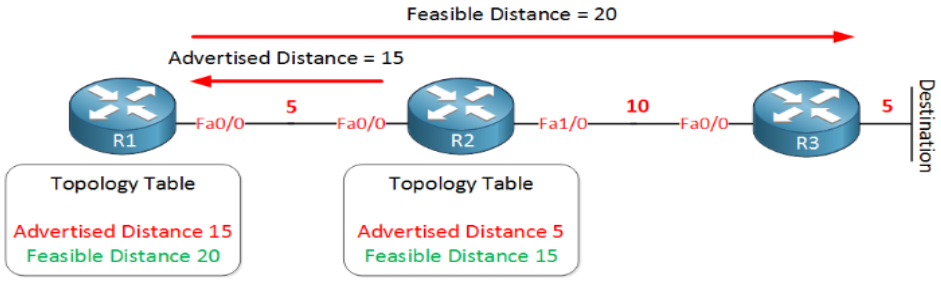
១០-៦-សេចក្តីផ្តើមចំពោះ EIGRP

EIGRP (Enhanced Interior Gateway Routing Protocol) ដែលជា Routing Protocol របស់ Cisco ។ បើអ្នកមិនស្គាត់ជំនាញជាមួយ Distance vector និង RIP នោះទេ អ្នកត្រូវតែអាន RIP មុនពេលបន្តចូលទៅក្នុង EIGRP ។ EIGRP ត្រូវបានគេហៅថា Hybrid ឬជា Advanced distance vector protocol ហើយច្បាប់ដែលអនុវត្តចំពោះ RIP ក៏ត្រូវបានអនុវត្តនៅទីនេះដែរមានដូចជា:

- Split Horizon
- Route Poisoning
- Poison Reverse

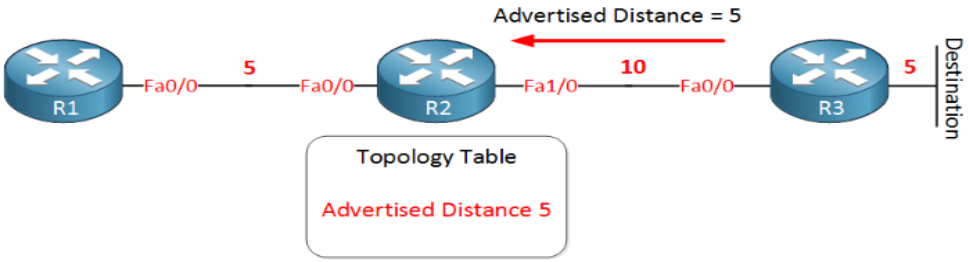
EIGRP routers និងចាប់ផ្តើមបញ្ជូននូវ Hello Packets ទៅត្បៃ Router ផ្សេងៗទៀតដូចជា OSPF ធ្វើការដែរ។ បើអ្នកបញ្ជូននូវ Hello Packets ហើយអ្នកទទួលវាមកវិញ អ្នកនិងក្លាយជាអ្នកជិតខាងរបស់វា ។

EIGRP neighbors និងផ្លាស់ប្តូរនូវព័ត៌មានអំពី Routing ដែលនិងត្រូវបាន Save នៅក្នុងតារាងរបស់ Topology ។ ផ្លូវដែលល្អប្រសើរបំផុតពីតារាងនៃ Topology និងត្រូវបានចំលងចូលទៅក្នុង routing table ។



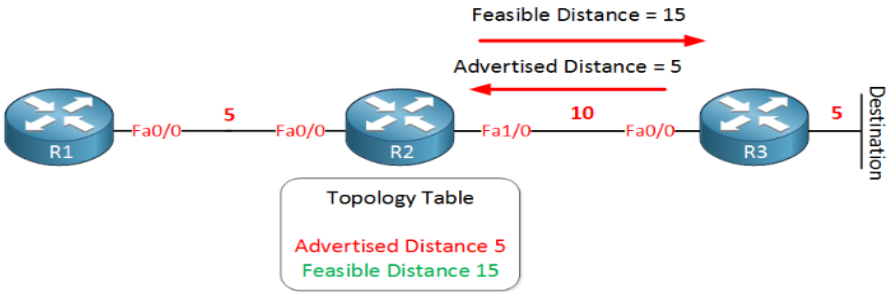
យើងមាន routers ចំនួន៣មានឈ្មោះថា R1,R2 និង R3 ។ យើងនិងគណនាចំពោះផ្លូវដែលប្រសើរជាងគេបំផុតមកកាន់គោលដៅដែលនៅខាងក្រោយ R3 ។ EIGRP ប្រើនូវសំនុំនៃ metrics មានដូចជា **bandwidth, delay, load and reliability** ដែលយើងនិងពិភាក្សានៅពេលក្រោយ។ តម្លៃទាំងនេះនិងមានរូបមន្តហើយ link នីមួយៗនិងត្រូវបានកំនត់ឲ្យនូវ metric មួយ។ មេត្រិកដែលមានតម្លៃទាបជាងគេគឺប្រសើរជាងគេ។

នៅក្នុងរូបភាពខាងលើយើងបានកំនត់ទៅលើ interfaces បើអ្នកពិនិត្យទៅលើ EIGRP router ពិតប្រាកដ អ្នកនិង



ឃើញនូវលេខដែលមានតម្លៃខ្ពស់បំផុតនិងពិបាកធ្វើការជាមួយ។

R3 និងផ្សព្វផ្សាយព័ត៌មានទៅឲ្យ R2 នូវមេត្រិករបស់វាមកកាន់គោលដៅ។ ជាមូលដ្ឋានគ្រឹះ R3 កំពុងនិយាយមកកាន់ R2 ថា “វាមានតម្លៃ 5 ដើម្បីទៅដល់ទីនោះ” ។ នេះត្រូវបានគេហៅថា advertised distance។ R2 មានតារាងនៃ topology និងនៅក្នុងតារាងនៃ topology វានិង Save នូវមេត្រិកនោះទុកគឺ៥ដែលប្រើសម្រាប់មកដល់គោលដៅ។ យើងមិនបានធ្វើវារួចរាល់នៅឡើយនោះទេដោយសារតែមានអ្វីដែល R2 និង Save នៅក្នុងតារាង



topology របស់វា។ យើងបានដឹងហើយថាចម្ងាយដែលបានផ្សព្វផ្សាយគឺ៥ ដោយសារតែ R3 បានប្រាប់យើង។ យើងក៏បានដឹងផងដែរថាមេត្រិកនៃ link រវាង R2 និង R3 ត្រូវបានភ្ជាប់ដោយផ្ទាល់។ R2 បានដឹងពីមេត្រិកសម្រាប់ផ្លូវសរុបមកកាន់គោលដៅដែលប្រវែងផ្លូវសរុបនេះគេហៅថា Feasible distance ហើយវាត្រូវបាន Save ទៅក្នុង topology table របស់វា។ អ្នកបានរៀនអំពីសញ្ញាណពីរសំខាន់នៃ EIGRP ។

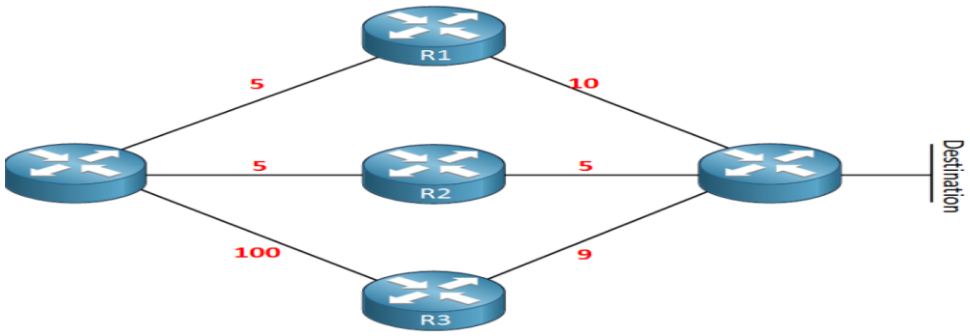
ចម្ងាយដែលបានផ្សព្វផ្សាយនេះដែលអ្នកជិតខាងរបស់អ្នកប្រាប់អ្នកពីរចម្ងាយដែលវាអាចមកដល់គោលដៅហើយ Feasible distance ដែលជាចម្ងាយសរុបរបស់អ្នកដែលត្រូវមកដល់គោលដៅ។ យើងមិនទាន់បានសម្រេចនៅឡើយពីព្រោះថា R1 ក៏កំពុងដំណើរការ EIGRP ដែរ។ R2 កំពុងបញ្ជូននូវ Feasible distance របស់វាមកកាន់ R1 គឺ១៥។ R1 និង Save នូវព័ត៌មាននេះនៅក្នុង topology table របស់វាជាចម្ងាយដែលផ្សព្វផ្សាយ។ R2 កំពុងប្រាប់ R1 ថាចម្ងាយគឺ១៥។

R1 ឥឡូវបានដឹងពីចម្ងាយពីគោលដៅគឺឆ្ងាយពី R2 និងយើងបានដឹងថាមេត្រិកសម្រាប់ Link រវាង R1 និង R2 ដែលវាអាចគណនាចំពោះចម្ងាយសរុបដែលត្រូវបានគេហៅថា Feasible distance ដែលព័ត៌មាននេះត្រូវបាន Save នៅក្នុង topology table។

សូមពិនិត្យទៅលើពាក្យបច្ចេកទេសទាំងនេះម្តងទៀត

- Advertised distance: គឺជាចម្ងាយពីគោលដៅទៅកាន់អ្នកជិតខាងរបស់អ្នក
- Feasible distance: គឺជាចម្ងាយសរុបមកកាន់គោលដៅ

ជាមួយ EIGRP ទោះបីជាយ៉ាងណាក៏ដោយត្រូវតែមានផ្លូវបម្រុងដែលគេហៅថា **feasible successor**។ តើយើងត្រូវរកវាឲ្យឃើញថាយើងមាន feasible successor តាមរបៀបណា?
សូមពិនិត្យមើល:



នៅក្នុងឧទាហរណ៍ងលើ យើងមាន Router ពីរដែលកំពុងដំណើរការ EIGRP ។ យើងកំពុងតែនៅខាង router ដែលគ្មានឈ្មោះនៅខាងឆ្វេងហើយយើងចង់ដឹងពីបញ្ហាពីរគឺ:

- តើផ្លូវមួយណាជា successor (ផ្លូវដែលប្រសើរជាងគេ)?
- តើយើងមាន feasible successors (backup paths) ដែរឬទេ?
- សូមបំពេញតារាងខាងក្រោមដើម្បីរកឲ្យឃើញ

	Advertised Distance	Feasible distance	
R1			
R2			
R3			

- បើអ្នកចង់សាកល្បងជាមួយជំនាញ EIGRP ត្រូវបំពេញនៅក្នុង advertised និង feasible distance ដោយខ្លួនអ្នកផ្ទាល់
- R1 កំពុងប្រាប់យើងថាគោលដៅគឺមានប្រវែង១០ R2 ប្រាបយើងថា ៥ និង R3 ប្រាប់យើងថា ៩។ យើងអាចបំពេញនៅក្នុង advertised distance នៃតារាង:

	Advertised Distance	Feasible distance	
R1	10		
R2	5		
R3	9		

- ដោយសារតែយើងបានដឹងពី Link ដែលបានភ្ជាប់ដោយផ្ទាល់ នោះយើងអាចបន្ថែមវាទៅក្នុង advertised distance ហើយបន្ទាបមកយើងបាន feasible distance.

	Advertised Distance	Feasible distance	
R1	10	15	
R2	5	10	
R3	9	109	

- ផ្លូវដែលមាន feasible distance តិចជាងគេបំផុតគឺជា successor(R2។ ដូច្នេះយើងបានឆ្លើយសំណួរទី ១

	Advertised Distance	Feasible distance	
R1	10	15	
R2	5	10	SUCCESSOR
R3	9	109	

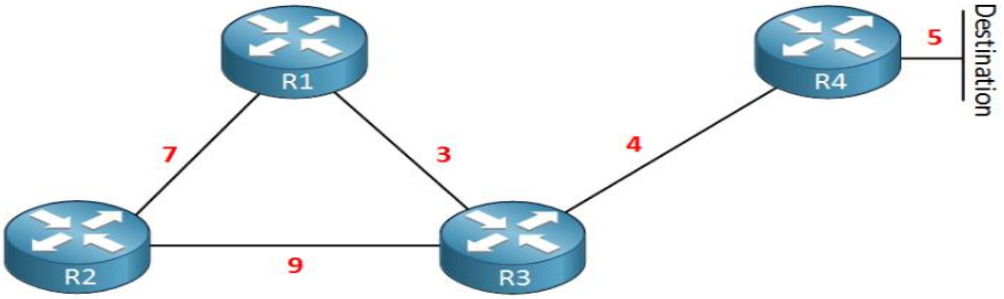
- អ្នកនិងរក Successor នៅក្នុង routing table
- ដើម្បីឆ្លើយសំនួរទី ២ យើងត្រូវរៀនពីរូបមន្ត
- ចម្ងាយផ្សព្វផ្សាយនៃ Feasible successor < Feasible distance of successor

- នេះគឺជាកន្លែងដែលខ្ញុំបានដឹង
- Router មួយអាចក្លាយជា backup path មួយបានបើវានៅជិតទៅនិងគោលដៅជាងប្រៀងសរុបនៃផ្លូវដែលល្អប្រសើរជាងគេបំផុត។
- សូមព្យាយាមនិងពិនិត្យបើ R1 ឬ R3 អាចជា Backup path
    - ចម្ងាយផ្សព្វផ្សាយនៃ R1 គឺ ១០ដែលស្មើទៅនិង Feasible distance នៃ R2 ដែលមានតម្លៃ ១០ដែរ។ វាត្រូវតែតូចជា ស្មើមិនល្អនោះទេ។ ដូច្នេះ R1 និងមិនអាចជា feasible successor នោះទេ
    - ចម្ងាយផ្សព្វផ្សាយនៃ R3 គឺ ៩ ដែលតូចជាចម្ងាយ feasible distance នៃ R2 ដែលមានតម្លៃគឺ ១០ ។ R3 និងជា feasible successor ហើយត្រូវបានប្រើជា backup path!

	Advertised Distance	Feasible distance	
R1	10	15	
R2	5	10	SUCCESSOR
R3	9	109	FEASIBLE SUCCESSOR

- វាជាការប្រសើរ។ ដូច្នេះ R2 គឺជា Sucessor របស់យើងហើយ R3 គឺជា Feasible successor មួយ។ អ្នកនិងរកឃើញថាការបញ្ចូលទៅក្នុង EIGRP toology ប៉ុន្តែយើងនិងរកឲ្យឃើញតែ successor នៅក្នុង routing table។ បើអ្នកបាត់បងនូវ Successor ដោយសារតែ Link បរាជ័យនោះ EIGRP និងចំលងឬ Paste នូវ feasible successor ចូលទៅក្នុង routing table។ នេះគឺជាអ្វីដែលធ្វើឲ្យ EIGRP ក្លាយជា FAST routing protocol ដែលមានល្បឿនលឿន ប៉ុន្តែបើអ្នកមាន feasible successor នៅក្នុង topology table
- ឥឡូវសូមពិនិត្យឲ្យល្អិតចំពោះ feasible distance នៃ R3 និង R1។ តើអ្នកឃើញអ្វីខ្លះ? មេត្រិកសម្រាប់ R3 នៅធ្លាយជាងចម្ងាយរបស់ R1។ តើវាមានន័យគ្រប់គ្រាន់ឬទេ? តើអ្នកដឹងទេថាវិស្វករ Cisco EIGRP ធ្វើឲ្យមានកំហុសដោយប្រើ non-optimal backup paths?
- ត្រូវចាំនៅក្នុងចិត្តថា EIGRP គឺជា distance vector protocol ។ វាមិនមែនជា Link State Protocol ដូចជា OSPF ដែលមានផែនទីនៃណេតវើកនោះទេ។
- Distance vector routing protocols ដឹងតែផ្លូវដែលត្រូវទៅ (Vector) និងចម្ងាយទៅគោលដៅ

- រូបមន្តដែលអ្នកត្រូវសម្រេចចិត្តថា EIGRP feasible successors គឺពីរបៀបដែល EIGRP អាចធានាថា



backup path គឺ 100% loop-free! ។ សូមពិនិត្យទៅលើឧទាហរណ៍

ត្រូវចាំនៅក្នុងចិត្តថា EIGRP គឺជាdistance vector routing protocol

យើងពិនិត្យទៅលើ EIGRP topology table របស់ R3 ហើយយើងចង់ឲ្យវាមកដល់គោលដៅខាងក្រោយ R4 ។ ត្រូវបំពេញតារាងខាងក្រោម:

	Advertised Distance	Feasible distance	
R4			
R1			
R2			

1. R4 និងផ្សព្វផ្សាយព័ត៌មានទៅគោលដៅ ណេតវើកមកកាន់ R3.
2. R3 និងផ្សព្វផ្សាយព័ត៌មានទៅគោលដៅ ណេតវើកមកកាន់ R1 និង R2.
3. R1 និងផ្សព្វផ្សាយមកកាន់ ណេតវើកមក R2
4. R2 និងផ្សព្វផ្សាយមកកាន់ ណេតវើកមកកាន់ R1
5. R1 និងផ្សព្វផ្សាយមកកាន់ ណេតវើកត្រឡប់មកកាន់ R3.
6. R2 និងផ្សព្វផ្សាយមកកាន់ ណេតវើកនេះត្រឡប់មកកាន់ R3

	Advertised Distance	Feasible distance	
R4	5		

R1	25		
R2	19		

យើងមាន distance ដែលបានផ្សព្វផ្សាយ។ អ្នកជិតខាងរបស់យើងកំពុងប្រាប់យើងឲ្យដឹងពីចម្ងាយដើម្បីឈានទៅដល់គោលដៅ ណេតវើក។ ដំហានបន្ទាប់ត្រូវបំពេញនៅក្នុង feasible distances។ តើយើងទទួលនូវលេខនៅក្នុងចម្ងាយផ្សព្វផ្សាយតាមរបៀបណា? សូមពិនិត្យទៅលើ Routers ចំនួន៣:

R4 គឺងាយណាស់។ គោលដៅមានចម្ងាយ 5 ដូចដែលបានឃើញនៅក្នុងរូបភាព។ វាត្រូវបានផ្សព្វផ្សាយទៅកាន់ R3 ហើយដាក់នៅក្នុង topology table។

R1 និងរៀនអំពីណេតវើកដែលជាគោលដៅតាមរយៈ R3 និង R2 ។ R3 និងផ្សព្វផ្សាយនូវចម្ងាយនៃ3+5+4=12 មកកាន់ R1។ R1 និងផ្សព្វផ្សាយបន្តនូវចម្ងាយមកកាន់ R3 ជា 3+5+4=12។ ហេតុអ្វីបានជាមិនដាក់ 12 នៅក្នុងផ្នែកនៃចម្ងាយដែលបានផ្សព្វផ្សាយក្នុងតារាង? វាជាសំណួរដ៏ល្អ។ ត្រូវចាំចំពោះ split-horizon ។ សូមកុំផ្សព្វផ្សាយនូវអ្វីដែលអ្នកបានរៀនទៅឲ្យអ្នកជិតខាងរបស់អ្នក។ R1 មិនផ្ញើនូវព័ត៌មានអំពី ណេតវើកនេះត្រឡប់មក R3 នោះទេ។ ឲ្យច្បាស់លាស់ “កុំបញ្ជូនអ្វីដែលអ្នកបានរៀននៅលើ interface នោះត្រឡប់មក interface នោះវិញដដែល” ។ តើអ្នកទទួលបាន២៥តាមរបៀបណា?
R4 និងផ្សព្វផ្សាយនូវចម្ងាយ ៥ មកកាន់ R3 ។ R3 និងផ្សព្វផ្សាយ 5+4=9 មកកាន់ R2 ។ R2 និងផ្សព្វផ្សាយ 5+4+9=18 មកកាន់ R1 ។ ជាចុងក្រោយ R1 និងផ្សព្វផ្សាយ 5+4+9+7=25 មកកាន់ R3 ។ Split Horizon មិនត្រូវបានអនុវត្តនៅទីនេះដោយសារតែ R1 បានរៀនអំពីគោលដៅនៅលើ Interface(R2) ។ បញ្ហាដូចគ្នាត្រូវបានអនុវត្តសម្រាប់ចម្ងាយដែលបានផ្សព្វផ្សាយនៃ 19 សម្រាប់ R2:

1. R4 ផ្សព្វផ្សាយចម្ងាយនៃ 5 មកកាន់ R3.
2. R3 ផ្សព្វផ្សាយចម្ងាយនៃ 5+4 = 9 មកកាន់ R1.
3. R1 ផ្សព្វផ្សាយចម្ងាយនៃ 5+4+3 = 12 មកកាន់ R2.
4. R2 ផ្សព្វផ្សាយចម្ងាយនៃ 5+4+3+7 = 19 មកកាន់ R3.

	Advertised Distance	Feasible distance	
R4	5	9	
R1	25	28	
R2	19	28	

R3 បានរៀនអំពីចម្ងាយដែលបានផ្សព្វផ្សាយពីអ្នកជិតខាងរបស់វាហើយដឹងអំពី Interface ដែលបានភ្ជាប់ដោយផ្ទាល់ ។ ដូច្នេះអ្នកអាចបំពេញនៅក្នុង feasible distance ។ ជំហានចុងក្រោយគឺជ្រើសរើសយក successor របស់យើង ។

	Advertised Distance	Feasible distance	
R4	5	9	SUCCESSOR
R1	25	28	
R2	19	28	

R4 មាន feasible distance ដែលតូចជាងគេបំផុត ។ ដូច្នេះវានិងក្លាយជា successor ។ សូមពិនិត្យមើលទៅលើ feasible successor និងពិនិត្យឲ្យដឹងថាវាមាន backup path:

Advertised distance of feasible successor < Feasible distance of successor.

ចម្ងាយដែលបានផ្សព្វផ្សាយនៃ R1 គឺ 25 និង R2 គឺ១៩ ដែលមានតម្លៃខ្ពស់ជាងតម្លៃ feasible distance នៃ R4 គឺ ៩ ។ ដូច្នេះវាមិនអាចក្លាយជា feasible successors នោះបានទេ ។ បើ Routers ទាំងនេះក្លាយជា backup path យើងនិងមាន loop កើតមានឡើង ។

បើអ្នកជិតខាងរបស់អ្នកនៅជិតទៅនិងគោលដៅជាផ្លូវសរុបនោះអ្នកយ៉ាងហោចណាស់ដឹងថាវាមិនទទួលអ្វីទាំងអស់មកកាន់គោលដៅដោយគ្រាន់តែបញ្ជូននូវ Packets តាមរយៈ Router របស់អ្នក ។ ប្រហែលវាមិនមែនជាផ្លូវដែលល្អប្រសើរបំផុតនោះទេ ប៉ុន្តែវាប្រាកដជា 100% loop-free! ។

បញ្ហាបន្ទាប់គឺចង់ឲ្យអ្នកដឹងអំពី EIGRP metrics ។ អ្នកបានដឹងថា RIP ប្រើ hop count ហើយ OSPF ប្រើតម្លៃនិង EIGRP គ្មានអ្វីក្រៅពីមេតរិកនោះទេ ។ មានវត្ថុ៤យ៉ាងគឺ:

- Bandwidth
- Delay
- Load
- Reliability

Bandwidth និង Delay គឺជាតម្លៃថេរ។ FastEthernet link គឺ 100 Mbit ហើយមានការពន្យាពេលគឺ 100usec(microseconds)។ Ethernet គឺ 10 Mbit ហើយមានការពន្យាគឺ 1000usec។ តម្លៃទាំងនេះមិនប្រែប្រួលនោះទេ។

Load គឺជាកម្រិតវល់នៃ Link ហើយ reliability គឺជាកម្រិតនៃការទុកចិត្តនៃ Link ដោយពិនិត្យទៅលើ Error ។ Load និង reliability គឺជាតម្លៃដែលជាប្រែប្រួលពីព្រោះថាវាប្រែប្រួលទៅតាមពេលវេលា។ តាមលំនាំដើម EIGRP អាចប្រើបានតែ bandwidth និង delay។ Load និង reliability ត្រូវបានបិទចោលតាមលំនាំដើម។ វាមានន័យគ្រប់គ្រាន់ពីព្រោះថាយើងចង់ឲ្យ routing protocols មានទំនុកចិត្ត។ អ្នកមិនចង់ឲ្យ EIGRP routers បញ្ជូន updates គ្រប់ពេលដែលធ្វើឲ្យ link ជាប់វល់រហូត។

១0-៧-សេចក្តីផ្តើមចំពោះ IS-IS

IS-IS គឺជា IGP(link-state routing protocol) ដែលដូចគ្នាទៅនិង OSPF។ វាបង្កើតនូវ neighbor adjacencies មាន areas ផ្លាស់ប្តូរនូវ link-state packets បង្កើតនូវ link-state database ហើយប្រើ Dijkstra SPF algorithm ដើម្បីស្វែងរកផ្លូវដែលប្រសើរបំផុតមកកាន់គោលដៅដែលត្រូវបានដំឡើងនៅក្នុង routing table។

ត្រឡប់មកក្រោយបន្តិចនៅពេលដែល OSPF និង IS-IS ត្រូវបានបង្កើតឡើង IP មិនត្រូវបានប្រើដូចសព្វថ្ងៃនោះទេ។ នៅពេលដែលមនុស្សគិតពី OSI ពួកគេគិតជាស្វ័យប្រវត្តទៅលើ *OSI-model* ។ ប៉ុន្តែ ISO (International Organization for Standardization) ក៏ត្រូវបានបង្កើតផងដែរដូចទៅនិង IP និង UDP ដែលគេហៅថា **CLNP (Connectionless-mode network Protocol) និង CLNS (Connectionless-mode network Service)។**

ISO ក៏ប្រើផងដែរនូវពាក្យបច្ចេកទេសខុសគ្នាដូចជា:

- Router = **Intermediate system**
- Host = End system

មិនដូចគ្នាទៅនិង OSPF ដែលត្រូវបានបង្កើតឡើងដោយ IETF (Internet Engineering Task Force)។ IS-IS ត្រូវបានបង្កើតឡើងដោយ **DEC សម្រាប់ CLNS**, មិនមែន IP។ ហេតុដូច្នេះហើយគេហៅថាជា IS-IS (Intermediate System – Intermediate System)។ នៅពេលក្រោយមក IS-IS ត្រូវបានអនុវត្ត។ ដូច្នេះវាគ៏អាចជា route IP ហើយបន្ទាប់មកគេហៅថា **integrated IS-IS**។

ឥឡូវនេះ យើងប្រើ IP នៅគ្រប់ទីកន្លែង។ ដូច្នេះអ្នកអាចឆ្ងល់ហេតុអ្វីបានជាគេចាប់អារម្មណ៍ទៅលើវា។ នៅពេលធ្វើការជាមួយ IS-IS អ្នកនិងឃើញពីអ្វីដែលយោងទៅកាន់ CLNP/CLNS នៅទីនេះ។ ឧទាហរណ៍ នៅពេលអ្នក Configure ទៅលើ router ID មួយ(គេហៅថា ណោតវើកEntity Title) វាត្រូវបានគេ Configure ជាមួយ **NSAP**

(**network Service Access Point Address**) ។ NSAP ប្រហាក់ប្រហែលទៅនិង IP address មួយដែលហើយវាមិនត្រូវបាន Configure ជាស្វ័យប្រវត្តនោះទេ ។ ដូច្នេះវាត្រូវតែយល់ពីការរៀបចំរបស់អ្នក ។

IS-IS វាស្ថិតនៅខាងលើ Ethernet header ដែលប្រើនូវ Header format ផ្ទាល់ខ្លួនរបស់វា ។ វាមិនត្រូវបាន encapsulate នៅក្នុង IP Packet ដូចជា Routing Protocol ផ្សេងៗទៀតនោះទេដូចជាOSPF និង EIGRP ។ IS-IS គឺជា Scalable routing protocol ដែលមានកម្រិតខ្ពស់ដែលត្រូវបានគេប្រើជាមួយ ISP ដែលមានទំហំធំ ។

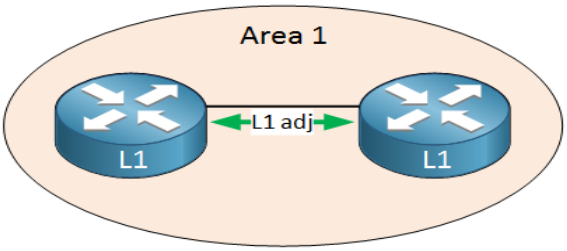
IS-IS ប្រើនូវតំបន់ដែល **entire router sits** ស្ថិតនៅក្នុង **area** មិនគ្រាន់តែជា interfaces របស់វាដូចជា OSPF នោះទេ ។ គ្មាន backbone area នោះទេ ។ Backbone ត្រូវបាន format ដោយ String នៃ routers ។ មាន Routers បីប្រភេទគឺ:

- **Level 1 system**: នេះគឺជា **intra-area router** ដែលវាស្គាល់តែអ្វីដែល local area មានហើយយើងនិងរៀននូវ prefixes ពី area របស់វា ។ វាបង្កើតនូវ level 1 link-state database និង SPF tree សម្រាប់ area
- **Level 2 system**: នេះគឺជា **backbone router** ដែលបានស្គាល់នូវគ្រប់ **intra-area ទាំងអស់និង inter-area routes** ។ វាបង្កើតនូវ level 2 link-state database និង SPF tree សម្រាប់ backbone
- **Level 1-2 system**: នេះគឺជា router មួយដែលសម្តែងនូវតួនាទីទាំងពីរ ។ វាបង្កើតនូវ level 1 និង 2 link-state database ជាច់ដោយឡែកពីគ្នានិង SPF trees ពីរ ។ មួយសម្រាប់ database នីមួយៗ ។

Level 1-2 គឺជា default នៅលើ Cisco IOS routers.

ដូចគ្នាទៅនិង Routing Protocol ផ្សេងៗទៀតដែរដូចជា OSPF និង EIGRP, IS-IS routers និងបញ្ជូននូវ hello packets ។ នៅពេលដែលអ្នកផ្ញើនិងទទួលលនូវ hello packets អ្នកនិងបង្កើតបាននូវអ្នកជិតខាងដែលនៅជាប់ៗគ្នា ។Routers និងបង្កើតបាន **neighbor adjacencies** ជាមួយ **routers** ដែលប្រើកម្រិតដូចគ្នា ។

សូមពិនិត្យមើលទៅលើឧទាហរណ៍វាមានតំបន់តែមួយ:



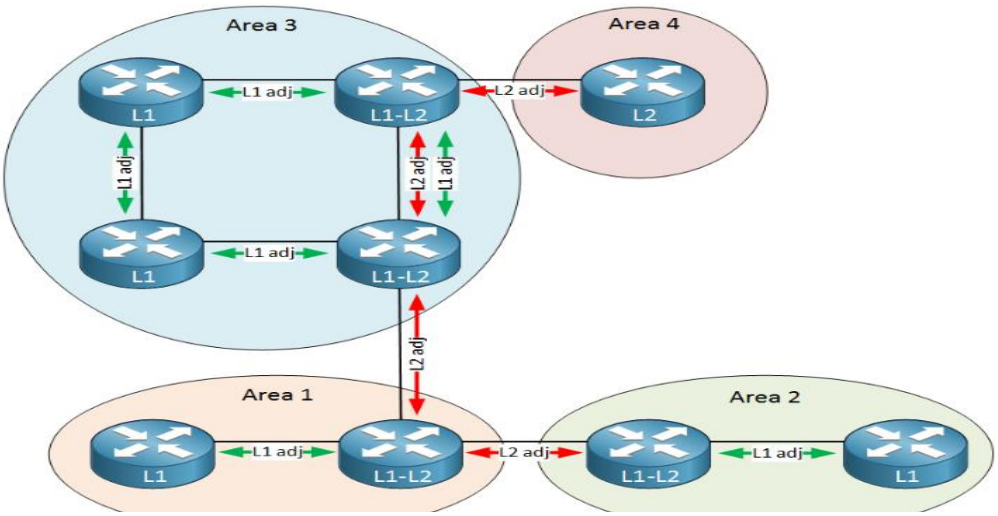
ខាងលើយើងមាន Routers ចំនួនពីរនៅក្នុងតំបន់តែមួយ ។ មានតំបន់តែមួយគត់ ។ ដូច្នេះ Routers ទាំងពីរនោះគឺត្រូវបាន Configure ជា Level 1 ។ Routers ទាំងពីរនិងបង្កើតបានជា level 1 neighbor ដែលនៅជាប់គ្នា ។

ឥឡូវនេះថែមតំបន់ទី២:

Level 1 routers ស្គាល់តែ local area ដែលវាស្ថិតនៅ ។ បើ level1 router ចង់ភ្ជាប់មកខាងក្រៅនៃតំបន់វាត្រូវតែប្រើ **level 2 router** ។ នៅក្នុងតំបន់នីមួយៗ យើង Configure router មួយមាន level 1-2 router ។ level 1-2 routers និងបង្កើតបាននូវអ្នកជិតខាងដែលនៅជាប់គ្នាចំនួនពីរ:

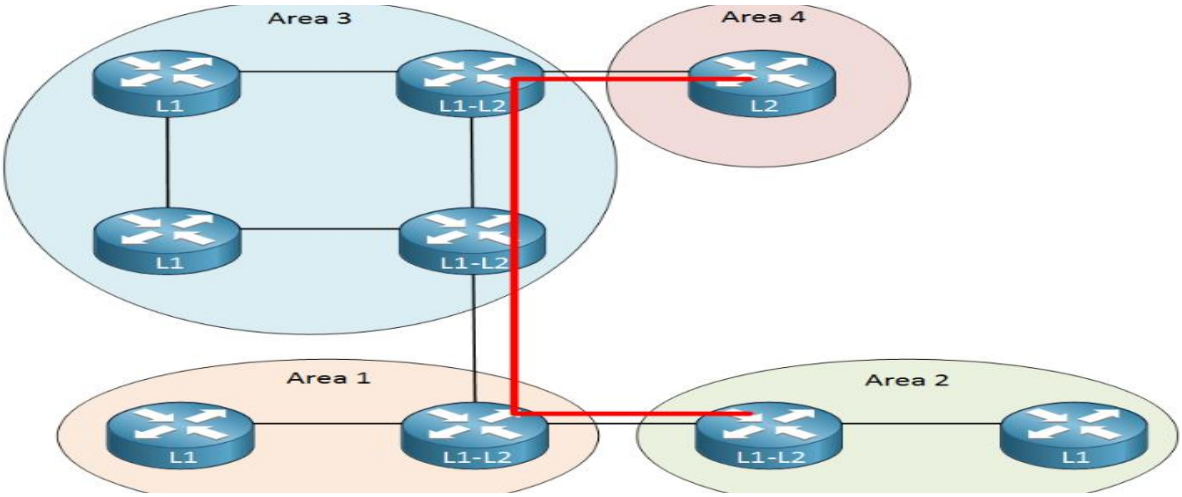
- Level 1 neighbor adjacency ជាមួយ router នៅក្នុងតំបន់ជាមួយគ្នា
- Level 2 neighbor adjacency ជាមួយ router នៅក្នុងតំបន់ផ្សេងគ្នា

នេះគឺជាឧទាហរណ៍មួយដែល topology ដ៏ធំមួយមានកម្រិតនៃ router levels ផ្សេងគ្នានិង adjacencies:



- ចំពោះ router នៅក្នុង area 4 គឺជា level 2 backbone router។ គ្មាន level 1 routers នៅក្នុង area 4 នោះទេ។ ដូច្នេះយើងមិនត្រូវការនូវ level 1-2 router នៅទីនោះទេ
- Area 3 មានពីរ level 1-2 routers។ Routers ទាំងនោះនិងបង្កើតបាននូវ neighbor adjacencies ពីរគឺ:
  - Level 1 adjacency
  - Level 2 adjacency

Router កម្រិតទាំងពីរបង្កើតបានជាខ្សែនៃ backbone routers:



LSPs (Link State Packets)

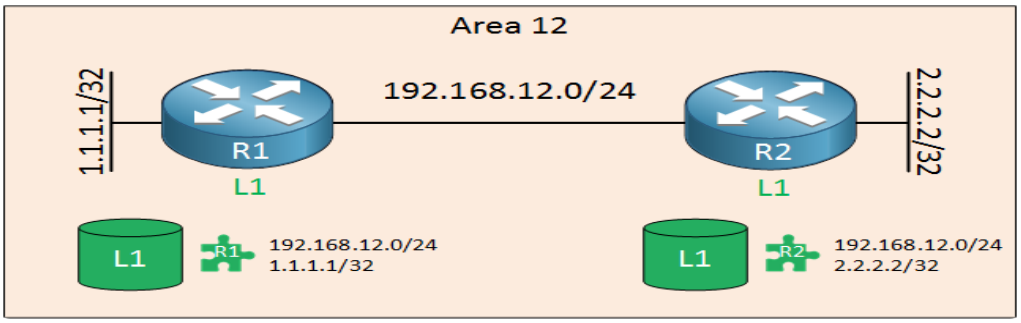
សូមពិភាក្សាអំពីដែល IS-IS routing ផ្លាស់ប្តូរព័ត៌មានគ្នាទៅវិញទៅមក។ វាប្រើ **LSPs(Link State Packet)** ដែលប្រហាក់ប្រហែលទៅនិង LSAs របស់ OSPF។ នៅក្នុង LSP អ្នកនិងរកឃើញថា:

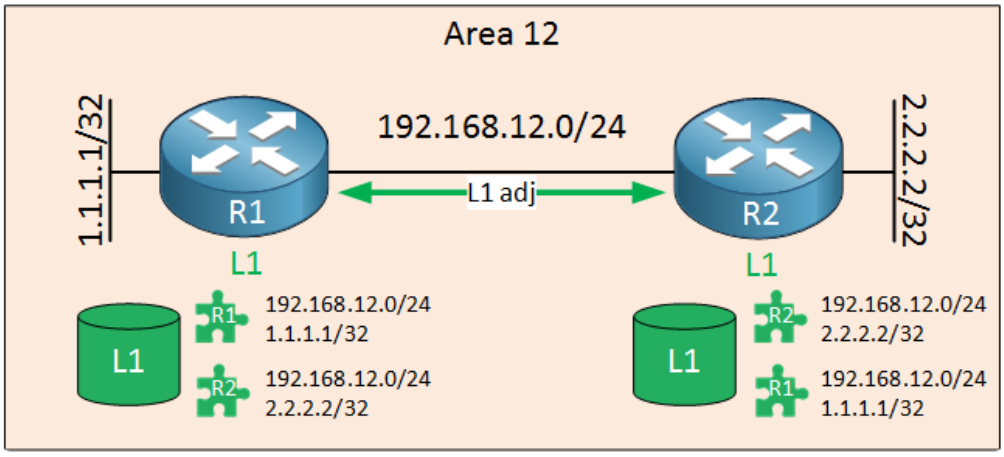
- One or more prefixes

- Adjacent neighbors
- Metric

សូមពិនិត្យមើលឲ្យបានជិតជិតពីរបៀបដែល IS-IS ប្រើ LSPs ដើម្បីផ្លាស់ប្តូរនូវព័ត៌មានអំពី routing។ ចាប់ផ្តើមជាមួយ Routers ពីរដែលត្រូវបាន Configure ឲ្យប្រើ IS-IS ប៉ុន្តែគ្មាន neighbor adjacency នោះទេ។

Router នីមួយៗនិងបង្កើតនូវ LSP(មានពណ៌បៃតង)។ នៅក្នុង LSP យើងរកឃើញថា ណេតវើកដែលបានភ្ជាប់ដោយផ្ទាល់ត្រូវបានផ្សព្វផ្សាយនៅក្នុង IS-IS។ ពីរបីវិនាទីក្រោយ routes ទាំងនោះនិងក្លាយជា neighbors។ R1 និង R2 ស្ថិតនៅក្នុងតំបន់តែមួយ។ ដូច្នេះវានិងបង្កើតបានជា level 1 neighbor adjacency។ Routers ទាំង

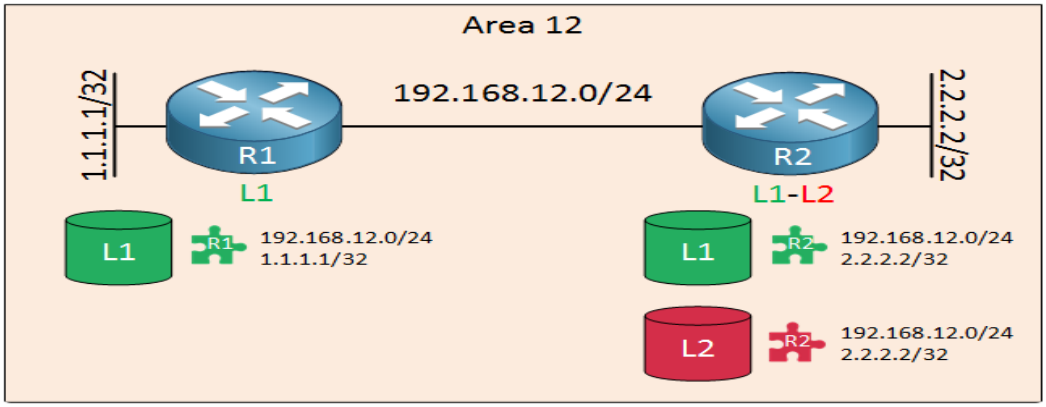




នេះនិង **flood នូវ LSPs** របស់គេនៅក្នុងតំបន់។ ដូច្នេះគ្រប់ Routers ទាំងអស់ដឹងអំពី LSPs ដែលមាននៅក្នុងតំបន់នោះ។ Routers ពីរបន្តែមនូវ LSP ចូលទៅក្នុង Database របស់វា។ Routers ទាំងនោះអាចប្រើនូវ SPF នៅលើ level 1 database ហើយ Configure ទៅលើផ្លូវដែលខ្លីជាងគេបំផុតទៅកាន់គោលដៅនីមួយៗ។

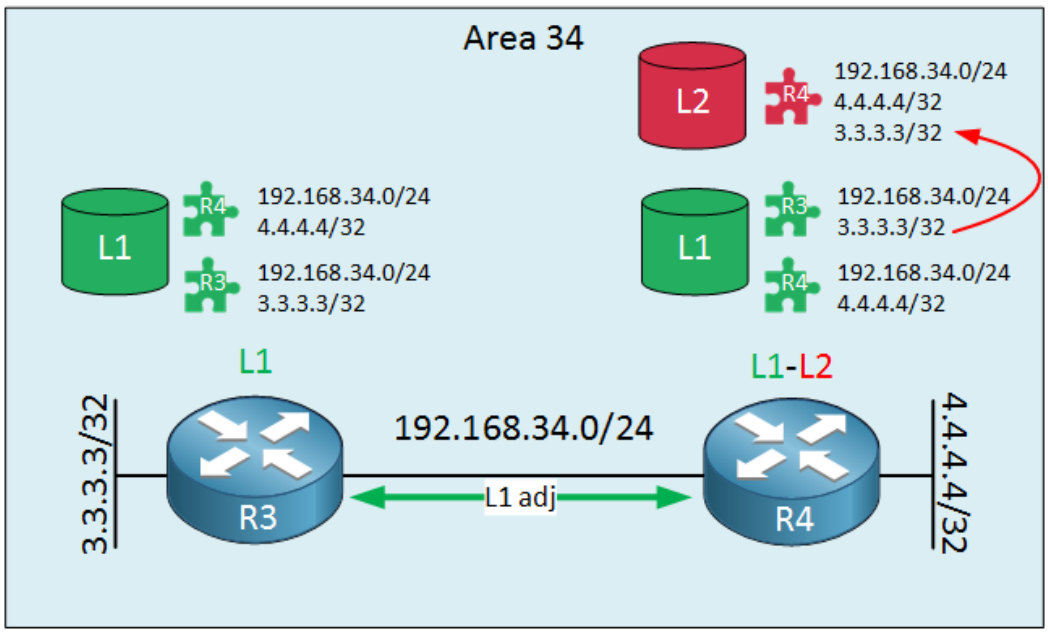
IS-IS ប្រើអ្វីមួយហៅថា DIS / Pseudonode ដែលប្រហាក់ប្រហែលទៅនិង OSPF's DR/BDR ដើម្បីកាត់បន្ថយនូវ flooding ដែលមិនត្រូវការ។

ឧបមាថាយើងចង់ភ្ជាប់ area 12 មកកាន់តំបន់ផ្សេងទៀត។ នេះមានន័យថាយើងត្រូវការនូវ level 2 router។ យើង



ប្តូរ R2 មកជា level 1-2 router។ ដូច្នេះយើងអាចឃើញអ្វីដែលអាចកើតមានឡើង។ នៅពេលនេះ យើងចាប់ផ្តើមជាមួយលក្ខណៈទទេរ មានន័យថាគ្មាន neighbor adjacency រវាង R1 និង R2 នោះទេ។ R2 មាននូវ Database ទី២គឺជា level 2 database។ ក្រៅពី level 1 database របស់វានិង level 1 LSP, វាក៏មាននូវ level 2 database នោះដែរ។ វាបង្កើតនូវ level 2 LSP និងគ្រប់ prefixes សម្រាប់ interfaces ត្រូវបានភ្ជាប់ដោយផ្ទាល់និងត្រូវបានផ្សព្វផ្សាយនៅក្នុង IS-IS។ IS-IS router នីមួយៗបង្កើតនូវ LSP តែមួយគត់សម្រាប់កម្រិតនីមួយៗ។ LSP នេះនាំយកនូវ prefixes ជាច្រើន។ ពីរបីវិនាទីក្រោយ R1 និង R2 បង្កើតបានជា level 1 neighbor adjacency:

ជាថ្មីម្តងទៀត R1 និង R2 និងផ្លាស់ប្តូរនូវ level 1 LSPs របស់គេ។ R2 ទទួលនូវ level 1 LSP ពី R1ហើយចតចំលងនូវ Prefixes ថ្មីពី level 1 database របស់វាទៅឲ្យ LSP នៅក្នុង level 2 database។ នៅក្នុងឧទាហរណ៍នេះគឺ1.1.1.1/32 ពី R1។ សូមបន្តទៅមុខទៀតជាមួយរឿងរ៉ាវនេះ។ យើងបន្ថែមនូវ Area ទី២ដូចគ្នាទៅនិង Area 12 ។ គ្មានការភ្ជាប់គ្នារវាងតំបន់ទាំងពីរនោះទេ ប៉ុន្តែ Routers ត្រូវបានបង្កើតឡើងជា level 1 neighbor adjacency នៅក្នុងតំបន់នោះ។



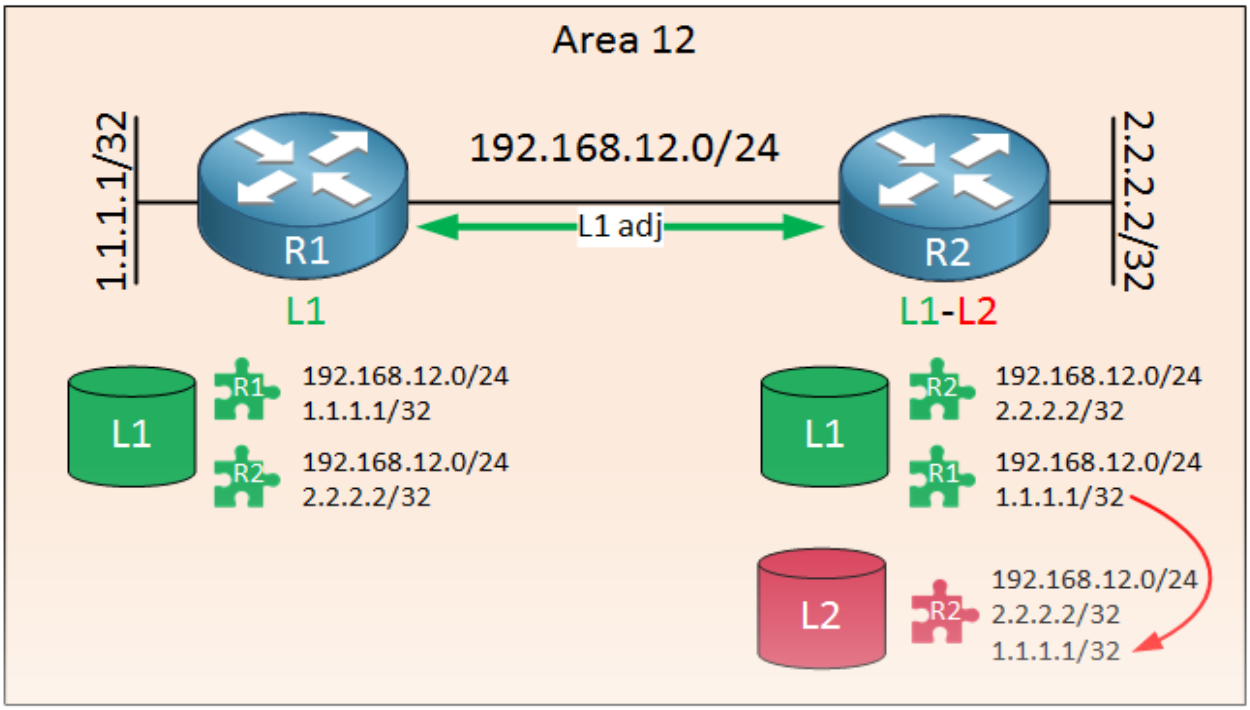
១០-៨-សេចក្តីផ្តើមចំពោះ BGP

BGP (Border Gateway Protocol) គឺជា Routing Protocol មួយដែលប្រើសម្រាប់ Internet ។ មុនពេលអ្នកបន្តទៅមុខ គួរម្លឹកឡើងវិញពី Routin Protocols ដូចជា RIP,OSPF និង EIGRP ជាដើម។ វាទាំងបីនេះគឺ Routin Protocols ដែលមានលក្ខណៈរួមគឺជា IGP(Interior Gateway Protocols) ។

យើងប្រើវាតែក្នុង autonomous system ប៉ុន្តែវាមិនសមរម្យសម្រាប់ប្រព័ន្ធ ណេតវើកធំៗដូចជា Internet នោះទេ។ RIP, OSPF និង EIGRP វាខុសគ្នា ប៉ុន្តែវាមានលក្ខណៈរួម។ គេប្រើវាសម្រាប់តែស្វែងរកផ្លូវដែលខ្លីជាងគេបំផុតមកកាន់គោលដៅ។

នៅពេលដែលយើងក្រឡេកទៅមើល Internet វិញ យើងមិនអាចស្វែងរកផ្លូវដែលខ្លីជាងគេយ៉ាងណានោះទេ?។ មាន Routing Protocol តែមួយគត់ដែលយើងប្រើនៅលើប្រព័ន្ធ Internet គឺ BGP។

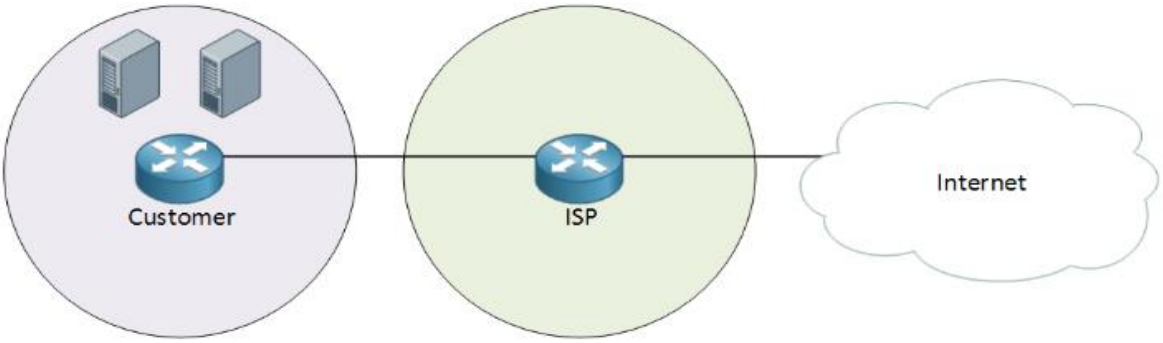
ហេតុដូចម្តេចបានជាត្រូវការ BGP?



សូមចាប់ផ្តើមជាមួយសេនារីយោដែលអ្នកអាចយល់បានដូចខាងក្រោម:

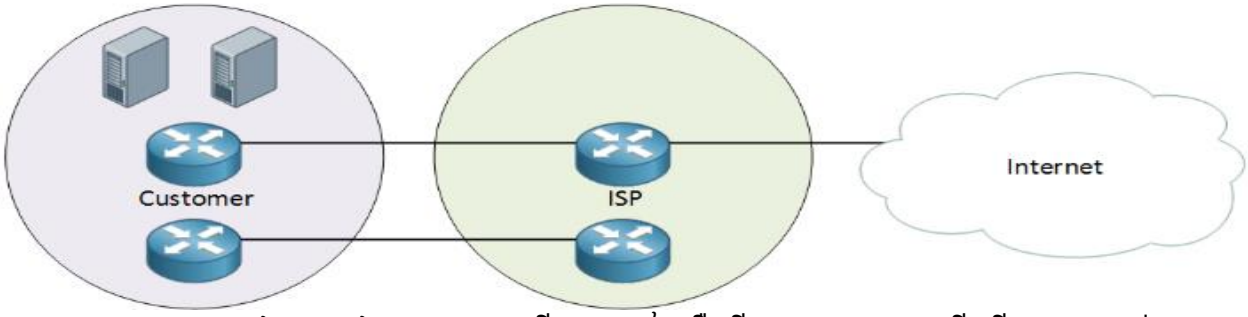
សព្វថ្ងៃនេះស្ទើរតែអ្វីៗទាំងអស់ត្រូវបានភ្ជាប់ទៅនិង Internet ។ នៅក្នុងរូបភាពខាងលើយើងមាន Customer ណេតវើកត្រូវបានភ្ជាប់ទៅនិង ISP ។ ISP របស់យើងត្រូវតែធ្វើឲ្យប្រើ Internet បាន។ ISP ផ្តល់ឲ្យនូវ Public IP address តែមួយគត់ដែលយើងអាចប្រើ Internet បាន។ ដើម្បីឲ្យប្រាកដថាអ្នកគ្រប់គ្នានៅក្នុង LAN នៅឯខាងអតិថិជនអាចប្រើ Internet បានយើងត្រូវប្រើ NAT/PAT ដើម្បីមកប្រែ IP address ឯកជនឲ្យត្រូវជាមួយ Public

IP address ។ នៅក្នុងរូបភាពនេះយើងមាន Client មួយដែលត្រូវការប្រើ Internet។ នៅខាង LAN របស់អតិថិជន ត្រូវការនូវ default route សម្រាប់ភ្ជាប់ទៅកាន់ ISP router ។ ចំពោះរូបភាពនេះយើងមិនត្រូវការ BGP នោះទេ។



នៅខាងអតិថិជនមាន Server ពីរដែលត្រូវការភ្ជាប់ទៅនិង Internet ។ វាអាចជា Mail server ឬ Web server ។

យើងប្រើ Port forwarding និង forward ចំពោះ Port ដែលត្រឹមត្រូវទៅកាន់ Server ទាំងនោះ ។ ដូច្នេះវានៅតែ



ត្រូវការនូវ IP address តែមួយដដែល។ មានជម្រើសមួយទៀតគឺប្រើ IP addresses ច្រើនពី ISP សម្រាប់ Server ទាំងនោះ។

ចុះបើយើងត្រូវការនូវ redundancy បន្ថែមទៀតតើធ្វើដូចម្តេច ? ការមានចំនុចបរាជ័យតែមួយមិនមែនជាចំណុចល្អ នោះទេ។ យើងអាចបន្ថែមនូវ router មួយទៀតនៅខាងអតិថិជនហើយភ្ជាប់វាទៅកាន់ ISP ។ អ្នកអាចប្រើ Link ទី ១សម្រាប់គ្រប់ចរាចរណ៍ទាំងអស់ហើយ link មួយទៀតសម្រាប់ Backup បាន។ យើងនៅតែមិនត្រូវការនូវ BGP ដដែលដោយគេអាចប្រើ default routing:

- ផ្សព្វផ្សាយ default route នៅក្នុង IGP នៅខាង Customer Router ទី១ជាមួយមេត្រិកទាបជាងគេ
- ផ្សព្វផ្សាយ default route នៅក្នុង IGP នៅលើ Customer router ទី២ជាមួយ metric ខ្ពស់

វាប្រាកដថា IGP បញ្ជូនគ្រប់ចរាចរណ៍ប្រើ Link ទី១ ។ នៅពេលដែល link បរាជ័យភ្លាម នោះ IGP និងធ្វើឲ្យ ចរាចរណ៍ទាំងអស់ត្រូវបានបញ្ជូនមកកាន់ Backup link វិញ ។

IGP របស់អ្នកនិងបញ្ជូននូវគ្រប់ចរាចរណ៍មកកាន់ Link ទី១និងមិនមកកាន់ Backup link នោះទេលុះត្រាតែមានភាពបរាជ័យ។ អ្នកអាចផ្សព្វផ្សាយ default route មួយដែលមានមេត្រិកដូចគ្នាបាន ប៉ុន្តែអ្នកនៅតែមាន 50/50% នៃការចែកគ្នា។ ចុះបើអ្នកចង់បានបញ្ជូន 80% នៃចរាចរណ៍ចេញក្រៅនៅលើ Link ទី១និង ២0%នៅលើ backup link វិញ? នោះវាមិនអាចកើតមានឡើងនោះទេ ប៉ុន្តែជាមួយ BGP អាចធ្វើទៅបាន។

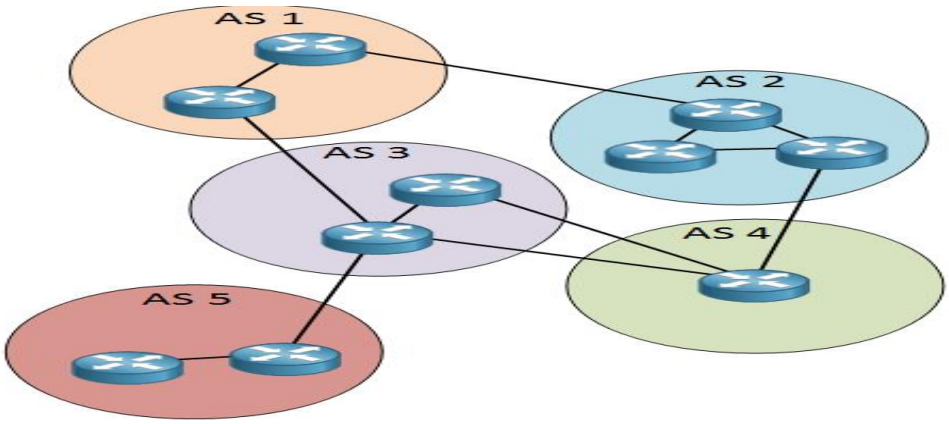
ក្រៅពីភ្ជាប់ទៅ ISP តែមួយ យើងអាចមាន ISP ពីរផ្សេងគ្នាបាន។ សម្រាប់ហេតុផលទល្យភាពវាមានសារៈសំខានណាស់ដែលយើងត្រូវមាន ISPs ពីរផ្សេងគ្នាដែលនៅក្នុងករណីនេះបើមួយខូចនោះយើងមាន Backup ISP សម្រាប់ប្រើវា។តើចំពោះ customer ណេតវើកយ៉ាងណាដែរ? យើងនៅតែមាន Servers ពីរដែលភ្ជាប់ទៅនិង Internet។

នៅក្នុងឧទាហរណ៍មុនយើងទទួលបាន public IP addresses ពី ISP ។ ឥឡូវនេះយើងភ្ជាប់ទៅកាន់ ISP ពីរផ្សេងគ្នា។ តើ Public IP address មួយណាដែលគួរប្រើ? ពី ISP1 ឬ ISP2? បើយើងប្រើ Public IP address ពី ISP1 ឬ ISP2 នោះ Servers ទាំងពីរនិងមិនអាច Access ទៅកាន់ ISP បានទេ។

ជំនួសឲ្យការប្រើ public IP address ពី ISP យើងទទួលបាននូវ public IP address ។ IP address space ដែលគ្រប់គ្រងដោយ IANA (Internet Assigned Numbers Authority – http://www.iana.org/) ។ IANA កំនត់នូវ IP address space ទៅឲ្យ Reginal Internet Registries ដូចជា RIPE ឬ ARIN ។ ការកំនត់នៃ IP address space ទៅឲ្យ ISPs ឬអង្គការធំៗ។ នៅពេលយើងទទួលបាននូវ Public IP address space បន្ទាប់មកយើងបញ្ជូនវាទៅឲ្យ ISP របស់យើង។ ការផ្សព្វផ្សាយនេះធ្វើទៅបានជាមួយ routing protocol ដែលជា BGP។

Autonomous Systems

ក្រៅពីការទទួលបាននូវ public IP address space យើងក៏គិតពី **AS (Autonomous System)**:



AS គឺជាបណ្តុំនៃ ណេតវើកដែលស្ថិតនៅក្រោម Domain នៃការគ្រប់គ្រងតែមួយ។ Internet គ្មានអ្វីក្រៅពីបណ្តុំនៃ autonomous system ដែលត្រូវបានភ្ជាប់ជាមួយគ្នាទៅវិញទៅមក។ នៅក្នុង autonomous systems យើងប្រើ IGP ដូចជា OSPF ឬ EIGRP។

ចំពោះ routing រវាង autonomous system ផ្សេងគ្នាយើងប្រើ EGP (**external gateway protocol**)។ សព្វថ្ងៃនេះមានតែ EGP ដែលយើងប្រើគឺ BGP។ តើយើងទទួលបាននូវលេខ autonomous system តាមរបៀបណា? វាដូចទៅនិង public IP address space ដែលអ្នកត្រូវចុះបញ្ជី។

លេខ Autonomous system គឺមានប្រវែង 16-bit ដែលមានន័យថាយើងមានចំនួន 65535 ដែលអាចជ្រើសរើសយកបាន។ វាដូចទៅនិង private និង public IP addresses គឺយើងមានបណ្តុំនៃ public និង private AS numbers។

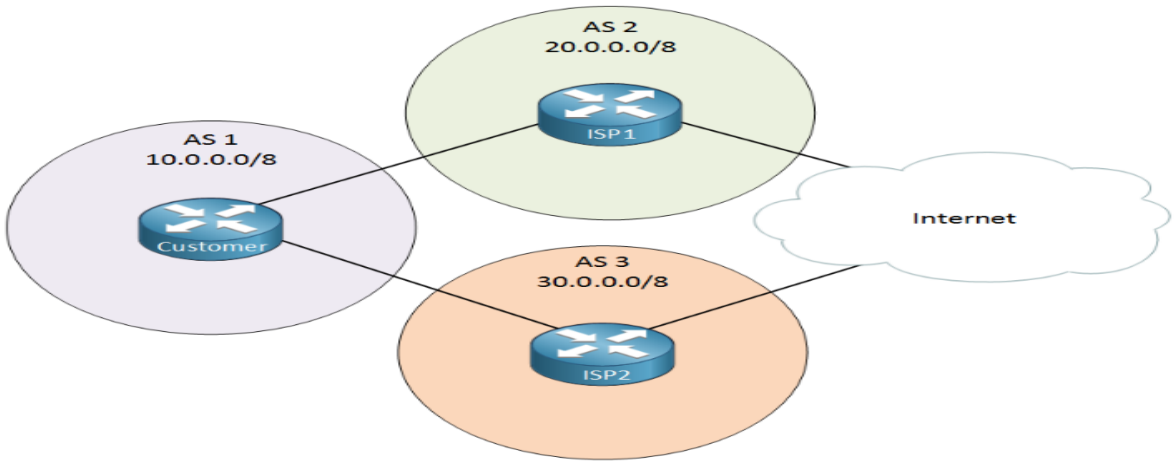
បណ្តុំទី១គឺ 1 – 64511 ដែលជាលេខ AS តែមួយគត់ហើយបណ្តុំចាប់ពី 64512 – 65535 គឺជាលេខ autonomous system ឯកជន។

UltraTools AS Information Lookup
BGP មានជម្រើសពីរគឺ:

- External BGP: ត្រូវបានប្រើរវាង autonomous systems
- Internal BGP: ត្រូវបានប្រើនៅក្នុង autonomous system.

External BGP គឺត្រូវបានប្រើដើម្បីផ្លាស់ប្តូរនូវព័ត៌មានអំពី routing រវាង autonomous system ផ្សេងគ្នា។ ឥឡូវនេះអ្នកមានគំនិតអំពីហេតុអ្វីបានជាត្រូវការ BGP និងអ្វីជា autonomous systems ។ Internet គឺជាកន្លែងដ៏ធំមួយដូចអ្វីដែលបានសរសេរនៅក្នុងសៀវភៅនេះគឺមានច្រើនជាង 500.000 prefixes ស្ថិតនៅក្នុង Internet routing table។
CIDR Report

នៅលើ Internet មាន Servers ជាច្រើន។ ចំពោះ Routers ត្រូវបានគេប្រើជាសាធារៈហើយអ្នកអាចប្រើវា

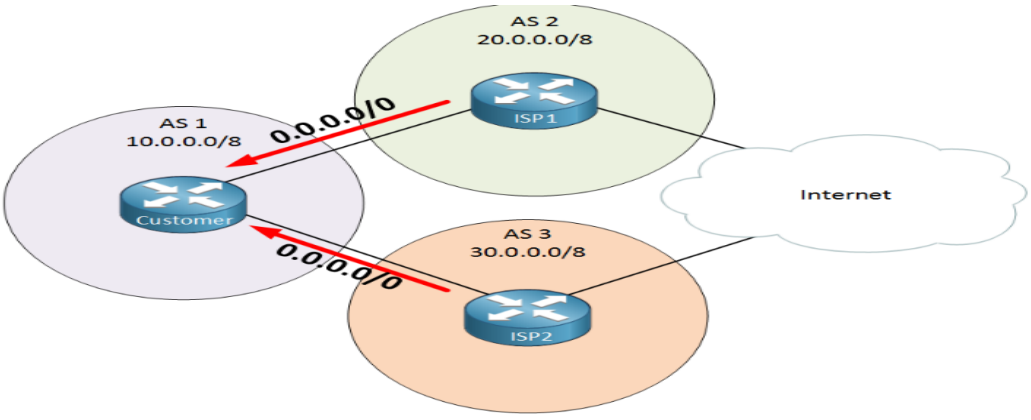


ដើម្បីពិនិត្យមើលពី Internet routing table។ បើអ្នកចង់មើលវាអ្នកអាចប្រើបញ្ជា **show ip route** និង **show ip bgp** ដើម្បីត្រួតពិនិត្យទៅលើ BGP ឬ routing table។ នៅពេលអ្នកដំណើរការ BGP ។ តើមានន័យថាយើងកំពុងមើល 500.000 prefixes នោះមែនទេ? សូមពិនិត្យមើលឧទាហរណ៍ខាងក្រោម:
ដូចនៅក្នុងរូបភាពខាងលើនៅក្នុង customer ណេតវើករបស់យើងមានលេខ autonomous system(AS 1) ហើយមាន IP address space (10.0.0.0 /8) ដោយចាត់ទុកថាវាជា public IP addresses។ យើងត្រូវបានភ្ជាប់ជាមួយ ISPs ពីរផ្សេងគ្នាហើយអ្នកអាចឃើញនូវលេខ AS របស់វាគឺ (AS2 and AS3) និង IP address space (20.0.0.0/8 and 30.0.0.0/8)។ យើងអាចភ្ជាប់ទៅនិង internet តាមរយៈ ISPs ទាំងពីរបាន។ យើងអាចប្រើ BGP ដើម្បីផ្សព្វាយចំពោះ address space របស់យើងមកកាន់ ISPs ។ ប៉ុន្តែតើ ISP និងផ្សព្វផ្សាយអ្វីទៅកាន់ customer តាមរយៈ BGP?

មានជម្រើសមួយចំនួនគឺ:

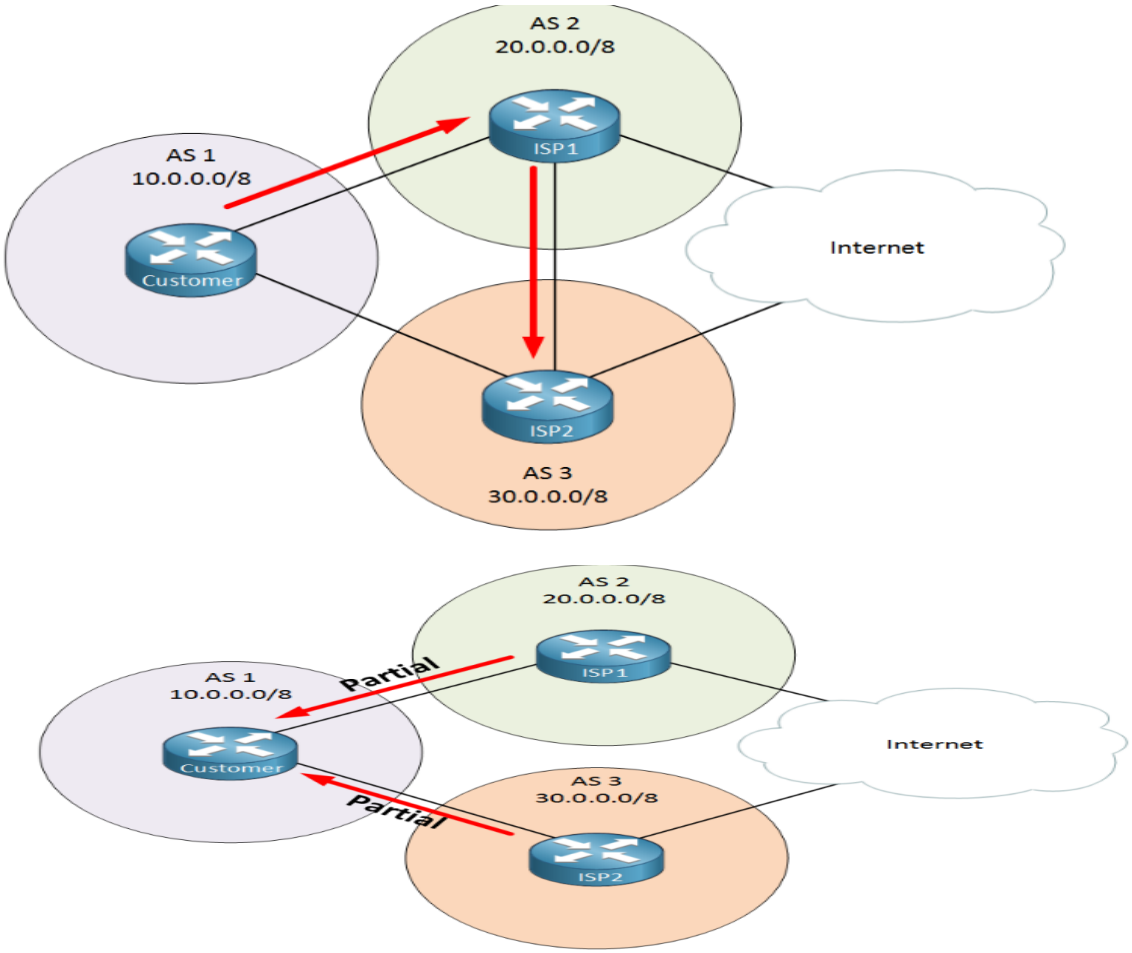
- វាផ្សព្វផ្សាយតែ Default route
- វាផ្សព្វផ្សាយតែ deault route និង routing table មួយផ្នែក
- វាផ្សព្វផ្សាយនូវ Internet routing table ទាំងមូល

Default Route



ការទទួលបាននូវ default route ត្រូវការនូវធនធានតិចតួចនៅលើ Router របស់អ្នកដោយហេតុថាអ្នកគ្រាន់តែមានការបញ្ចូលតែមួយគត់ដោយឈានទៅភ្ជាប់ណេតវើកខាងក្រៅបាន។ customer router និងផ្សព្វផ្សាយនូវ 10.0.0.0/8 ណេតវើករបស់វាទៅកាន់ ISPs ទាំងពីរដែលនិងផ្សព្វផ្សាយវាទៅកាន់ AS ផ្សេងទៀតដែលត្រូវបានភ្ជាប់ជាមួយហើយយើងប្រើ default route ដើម្បីភ្ជាប់មកកាន់ Internet។ ផ្នែកខាងក្រោមនៃការ Configure គឺថា customer ណេតវើកមិនបានដឹងអ្វីនៅពីក្រោយ ISP1 និង ISP2 នោះទេ។ យើងមានការភ្ជាប់ពីព្រោះថាមាន

default routes។ ប៉ុន្តែអាចឈានទៅរក sub-optimal routing។ បើយើងគ្រាន់តែមាន default routes ហើយ បន្ទាប់មកនិងបញ្ជូននូវគ្រប់ចរាចរណ៍ទៅឲ្យ ISP មួយក្នុងចំណោមវា។
នេះវាអាចកើតមានឡើងបើប្រើតែ default routes:

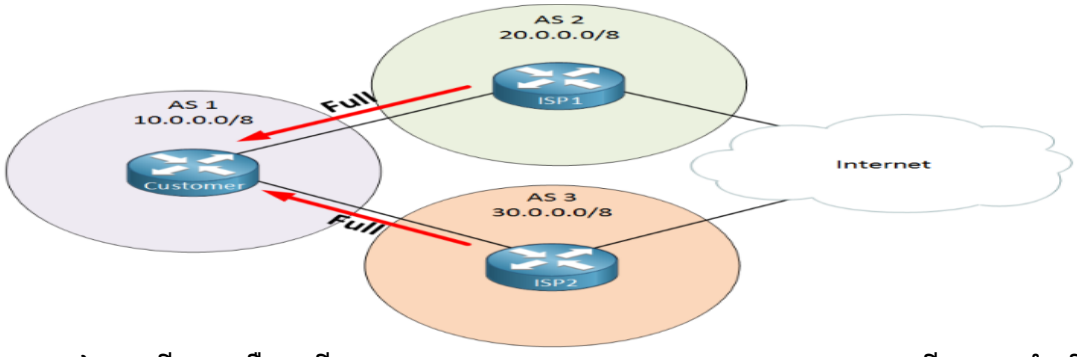


ចំពោះ customer ណោតវើកបានទទួលតែ default route ពី ISPs ទាំងពីរហើយយើងបានជ្រើសរើសដើម្បីប្រើ default route នៃ ISP1 ដើម្បីផ្ញើនូវចរាចរណ៍ចេញទាំងអស់ទៅ។ នេះមានន័យថានៅពេលណាក៏ដោយដែលយើង បញ្ជូនចរាចរណ៍សម្រាប់ 30.0.0.0 /8 (ISP2) វានិងត្រូវបានផ្ញើទៅកាន់ ISP1 ហើយបន្ទាប់មក ISP2។

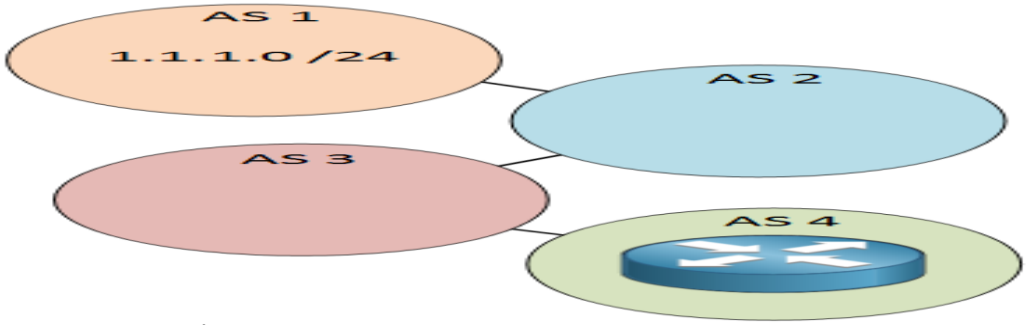
ដំណើរការនៃការធ្វើបច្ចុប្បន្នភាពនៃ Routing ជាផ្នែក

យើងក៏ទទួលបានផងដែរនូវ **partial routing table** បន្ថែមនូវ default route។ ការធ្វើបច្ចុប្បន្នភាពជាផ្នែកអាចរួម មានគ្រប់ IP address space ដែល ISPs មានកំនត់ទៅឲ្យអតិថិជនរបស់គេ។

Internet Routing Table ពេញលេញ



ជម្រើសចុងក្រោយដែលយើងមានគឺថាយើងទទួលបាននូវ Internet routing table ពេញលេញពី ISPs ទាំងពីរ។



វាទាមទារនូវធនធានបន្ថែមច្រើនទៀត។ ប៉ុន្តែយើងនិងធ្វើឲ្យការសម្រេចចិត្តជ្រើសរើសផ្លូវបានល្អប្រសើរ

Path Vector

BGP ត្រូវបានគេហៅថាជា path vector routing protocol មួយ។ តើវាមានន័យដូចម្តេច?

សូមពិនិត្យមើលរូបភាពនេះ

យើងមាន autonomous systems ចំនួន៤ហើយយើងកំពុងដំណើរការនូវ BGP ដើម្បីផ្លាស់ប្តូរនូវព័ត៌មានអំពី routing ។ នៅក្នុង AS 1 មានណេតវើក1.1.1.0 /24 ហើយវាត្រូវបានផ្សព្វផ្សាយទៅកាន់ AS 2, AS 3 និង AS 4។

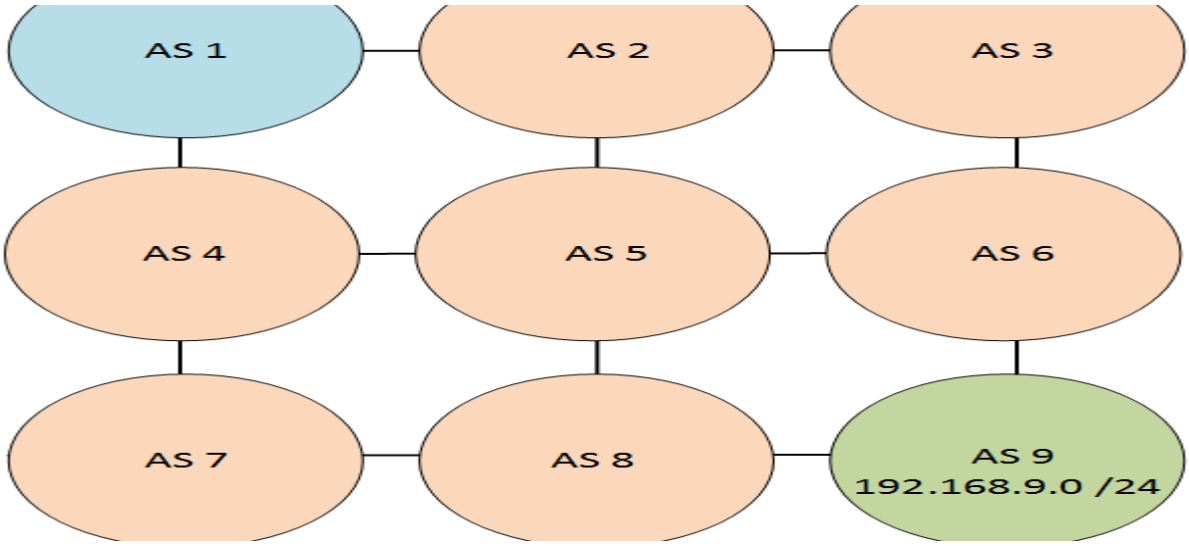
បើយើងពិនិត្យមើលទៅលើ BGP table នៃ Router នៅក្នុង AS4 ហើយបន្ទាប់មកយើងពិនិត្យ ណេតវើក 1.1.1.0 /24 ប៉ុន្តែវាក៏បានរក្សាទុកនូវផ្លូវដែលយើងមានដើម្បីទៅដល់ទីនោះ។ វានិងរក្សាទុកនូវ prefix ប៉ុន្តែក៏រក្សាទុកនូវផ្លូវដែលវាមានក្នុងគោលបំណងដើម្បីទៅដល់ 1.1.1.0 /24។ នេះគឺជាឧទាហរណ៍នៃ BGP router:

```
pntc>show ip bgp
BGP table version is 128380331, local router ID is 203.202.125.6
Status codes: s suppressed, d damped, h history,  * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, x best-external
Origin codes: i - IGP, e - EGP,  ? - incomplete
```

ណេតវើក	Next Hop	Metric LocPrf Weight Path
* 1.0.0.0/24	202.160.242.71	0 7473 15169 i

ដោយប្រើបញ្ជា **show ip bgp** យើងអាចពិនិត្យទៅលើ BGP table បានហើយឃើញថា Router នេះស្គាល់ពី ណេតវើក1.0.0.0 /24។ next-hop IP address គឺ 202.160.242.71។ នៅបន្ទាត់ចុងក្រោយអ្នកបានឃើញផ្លូវដែលមានលេខ 7473 15169។ វាគឺជា autonomous systems ដែលយើងត្រូវទៅដល់គោលដៅនៃណេតវើកនេះបាន។
ការជ្រើសរើសនៃ BGP Route

អ្វីដែល IGPs មានជារួមគឺថាវាសុទ្ធតែចង់ស្វែងរកផ្លូវដែលខ្លីជាងគេបំផុតដើម្បីទៅដល់គោលដៅ។



BGP ធ្វើការខុសគ្នាដោយសារតែ autonomous systems ជារបស់ ISPs ផ្សេងៗគ្នាឬអង្គការផ្សេងៗគ្នាដែលយើងចង់ឲ្យមានឥទ្ធិពលទៅលើ routing របស់យើង។

សូមពិនិត្យមើលទៅលើឧទាហរណ៍នេះ

BGP អនុញ្ញាតឲ្យយើងប្រើគោលការណ៍ routing នៅឯកម្រិតនៃ autonomous system។ នៅក្នុងរូបភាពខាងលើយើងមាន 9 autonomous systems ហើយនៅក្នុង AS 9 យើងមាន ណេតវើក192.168.9.0/24។ បើយើងពិនិត្យនឯ AS 1 បន្ទាប់មកមានផ្លូវជាច្រើនផ្សេងៗគ្នាយើងអាចឈានទៅរក ណេតវើក192.168.9.0/24 នៅក្នុង AS 9។ តើនេះមានន័យថា ណេតវើកadministrator នៅឯ AS 1 អាចជ្រើសរើសនូវផ្លូវដែលយើងត្រូវប្រើឬ?

មិនមែននោះទេដោយសារតែហេតុផលខាងក្រោម:

- អ្នកអាចជ្រើសរើសនូវ **exit path**...AS1 ដែលអាចធ្វើនូវចរាចរណ៍ទៅកាន់ AS 2 ឬ AS4 ប៉ុន្តែអ្នកមិនអាចធ្វើការសម្រេចចិត្តចំពោះ routing សម្រាប់ autonomous systems ផ្សេងនោះទេ

- ចំពោះ autonomous system នីមួយៗនិងផ្សព្វផ្សាយតែផ្លូវដែលល្អប្រសើរបំផុតទៅកាន់ autonomous system។ AS 1 និងរៀនអំពីផ្លូវដែលល្អប្រសើរជាងគេបំផុតពី AS 2 និង AS 4 លុះត្រាតែផ្លូវដែលប្រសើរជាងគេបរាជ័យហើយបន្ទាប់មកអ្នកនិងរៀនអំពីផ្លូវដែលប្រសើរជាងគេទី២

១0-៩-សេចក្តីផ្តើមនៃ Frame-Relay

Frame-relay គឺជា protocol មួយនៃ WAN protocol ដែលអ្នកត្រូវស្វែងយល់។ មុនពេលចាប់ផ្តើម

ជាមួយ frame relay សូមពិនិត្យមើលទៅលើរឿងដ៏តូចមួយ។ យើងមាន ណេតវើកដែលមាន៤ទីតាំង:

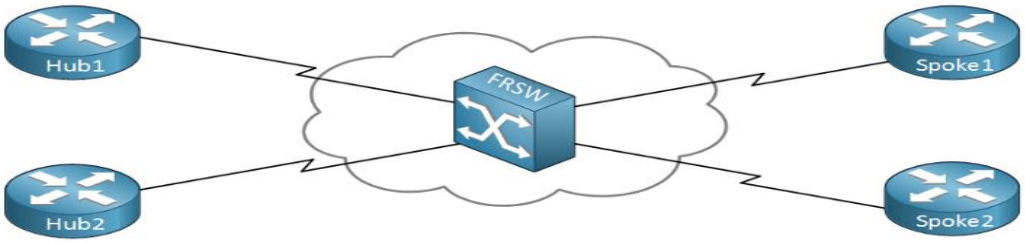
មាន R1, R2, R3 និង R4។ ដោយសារតែយើងភ្ជាប់រវាងទីតាំងយើងមាន ISP ដែលបានលក់ឲ្យយើងនូវ leased lines ចំនួនបី:

- រវាង R1 និង R2
- រវាង R1 និង R3
- រវាង R1 និង R4

ការប្រើ leased lines គឺអស្ចារ្យ។ អ្នកគឺជាមនុស្សតែម្នាក់គត់ដែលប្រើនូវខ្សែទាំងនោះដោយសារតែជាអ្នកចំណាយលុយទៅលើវា។ នេះមានន័យថាវាមានគុណភាពខ្ពស់ហើយមានការកកស្ទះកម្រិតទាប(បើអ្នកមានបណ្តាញដែលមានល្បឿនលឿន)។ វាមានសុវត្ថិភាពខ្ពស់ពីព្រោះមានតែចរាចរណ៍របស់អ្នកប៉ុណ្ណោះដែលមានឆ្លងកាត់តាមខ្សែទាំងនោះ។ មានកត្តាមួយចំនួនដែលអាចធ្វើឲ្យមានការធ្លាក់ចុះនៃបណ្តាញទាំងនោះ

- មានអស់ថ្លៃច្រើន
- នៅលើ R1 អ្នកនិងត្រូវការនូវ Interace ចំនួន៣សម្រាប់ Leased line នីមួយៗ។ ត្រូវ Interface កាន់តែច្រើន នោះការចំណាយក៏កាន់តែច្រើនដែរ
- តើមានអ្វីកើតឡើងបើអ្នកកំពុងបំលាស់ទីទៅកាន់ site R1? អ្នកមិនអាចបំលាស់ទីជាមួយ leased lines បាននោះទេ

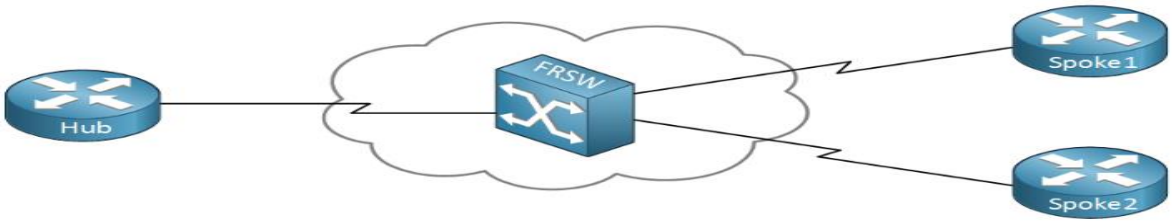
- ចុះបើ R1 ខូចវិញ ? R2,R3 និង R4 និងត្រូវស្ថិតនៅឯកោដែរ



នៅក្នុងរូបភាពនៃ Frame relay មួយហើយវាដំណើរការខុសគ្នាបន្តិច។ គំនិតនៅខាងក្រោយនៃ frame relay គឺថា អ្នកមាននូវហេដ្ឋារចនាសម្ព័ន្ធមួយពី ISP និងមានអតិថិជនជាច្រើនភ្ជាប់ជាមួយវាដោយចែកចាយនូវអ្វីៗទាំងអស់។

នៅកណ្តាលអ្នកឃើញពពកដែលអ្នកមិនធ្លាប់បានឃើញពីមុនមក។ រូបភាព Icon នេះគែជា frame relay switch។ ពពកគេហៅថា **frame relay cloud** ហើយហេតុផលដែលគេហៅបែបនេះពីព្រោះវាមានអតិថិជនច្រើន ហើយមិនបានដឹងថាវាមានអ្វីកើតមានឡើងនៅក្នុងពពកនោះ។

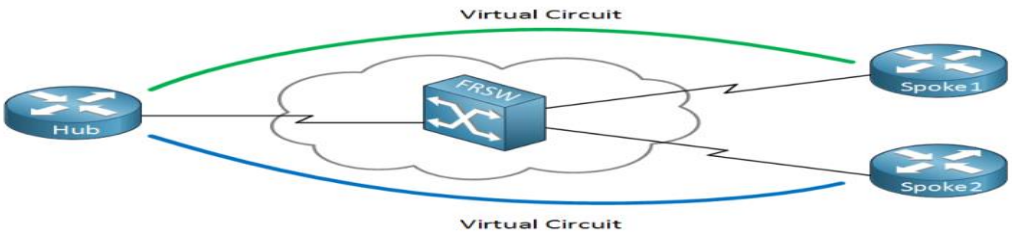
តើអ្នកឃើញអ្វី? មានអតិថិជនពីរ(១និង២)ហើយវាមាន Headquarters (Hub) ហើយការិយាល័យ



សាខាគឺ Spoke។ រូបភាពមួយទៀតគឺ frame relay ណេតវើកដែលមាន routers ចំនួនបីពីក្រុមហ៊ុនមួយ:

មាន router មួយនៅឯ headquarters (Hub)ហើយយើងមានសាខាពីរគឺSpoke1 និង Spoke2។ វាទាំងពីរត្រូវបានភ្ជាប់ទៅនិង frame relay cloud។

យើងហៅវាជាអ្នកផ្តល់សេវាពីព្រោះយើងចង់មានការភ្ជាប់និងសំណួរទី១ហេតុអ្វីបានជាគេសួរថាតើភ្ជាប់ទីតាំងមួយណាដំបូង ? នៅក្នុងឧទាហរណ៍ខាងលើ អ្នកឃើញនូវ **virtual circuits** ចំនួនពីរគឺពណ៌បៃតងនិងពណ៌ខៀវ



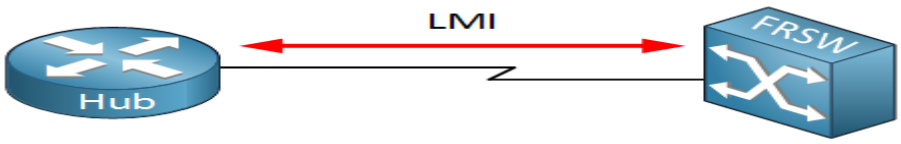
មួយ។

ជាមួយ frame relay មានភាពខុសគ្នារវាងការភ្ជាប់ physical និង logical។

ការភ្ជាប់ជា physical connection គឺគ្រានតែជា serial cable ដែលត្រូវបានភ្ជាប់ទៅនិងអ្នកផ្តល់សេវា។ចំពោះ logical links គឺជា virtual circuits។ ដូចដែលអ្នកបានឃើញថាមាន virtual circuit ពី Spoke1 ទៅកាន់ Hub ហើយមួយទៀតពី Spoke2 ទៅកាន់ Hub វិញ។ នេះមានន័យថាយើងអាចបញ្ជូននូវចរាចរណ៍តាមរយៈនៃ virtual circuits រវាង:

- Spoke1 និង Hub
- Spoke2 និង Hub

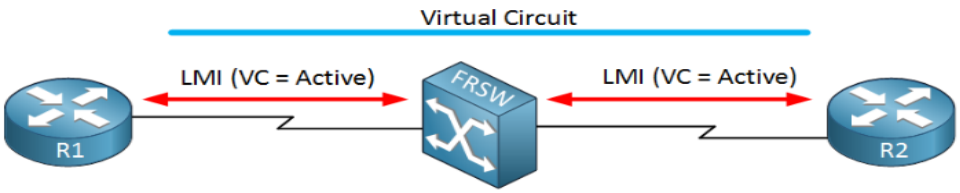
គ្មាន virtual circuit រវាង Spoke1 និង Spoke2។ តើមានន័យថាគ្មានការភ្ជាបរវាងវាមែនទេ? ទេ អ្នកនៅតែមានការភ្ជាប់នៅឡើយដោយបញ្ជូននូវទិន្នន័យមកកាន់ Hub។ តាមការពិតអ្នកអាចមាន virtual circuit មួយទៀតរវាង Spoke1 និង Spoke2 ។ បុន្តែអ្នកត្រូវចំណាយទៅលើវា។ Virtual circuits ត្រូវបានគេហៅថា **PVC (Permanent Virtual Circuit)** ។ អ្នកក៏ចំណាយទៅលើល្បឿនពិតប្រាកដដែលគេហៅថា **CIR (Committed Information Rate)** ។ បញ្ហាមួយរបស់ frame relay គឺថានៅពេលដែលគ្មានអតិថិជនប្រើ Frame relay ណេតវើកអ្នកទទួលបាននូវល្បឿនខ្ពស់ជាអ្វីដែលអ្នកបានចំណាយ។ ទោះបីជាយ៉ាងណាក៏ដោយ CIR គឺជាល្បឿនដែលត្រូវបានធានា។



តើអ្នកអាចដឹងថា PVC កំពុងដំណើរការឬទេតាមរបៀបណា?

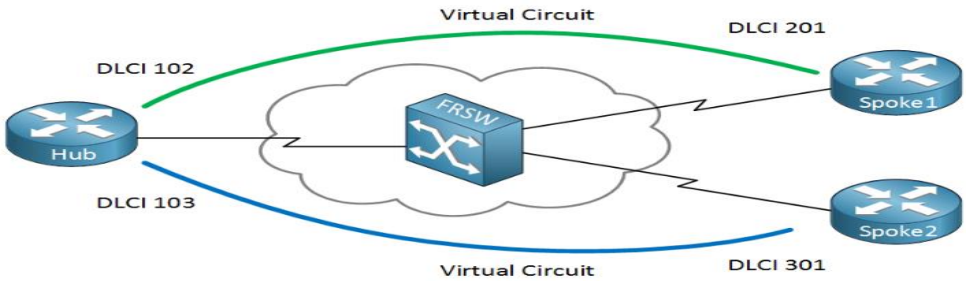
Frame-relay ប្រើអ្វីមួយហៅថា **LMI** ដែលមានពាក្យពេញថា **Local Management Interface**។ LMI មានតួនាទីពីរគឺ:

- វាជាយន្តការ Keepalive
- វាប្រាប់យើងថាបើ PVC មានសកម្មភាពឬអសកម្ម
- វាក៏ឲ្យយើងប្រើ DLCI (Data Link Connection Identifier)។ យើងត្រឡប់មក LMI ដែលមានបីប្រភេទ។ វាទាំងអសដូចគ្នា ប៉ុន្តែមានស្តង់ដាបីដែលមិនត្រូវគ្នា។ អ្វីដែលអ្នកជ្រើសរើសត្រូវប្រាកដថាវាដូចគ្នារវាងឧបករណ៍ពីរ:
- Cisco
- ANSI T1.617 Annex D
- ITU-T Q.933 Annex A



ដូច្នេះបើអ្នកយក Cisco នៅម្ខាង នោះត្រូវប្រើ Cisco នៅម្ខាងទៀតផងដែរ។

នេះគឺជាឧទាហរណ៍នៃ LMI នៅពេលមានសកម្មភាព។ នៅកណ្តាលយើងមាន frame relay switch។ LMI packets ត្រូវបានបញ្ជូនរវាង R1 និង frame relay switch ហើយ R2 និង frame relay switch។ frame relay switch ប្រាប់ពី routers របស់យើងថា PVC កំពុងមានសកម្មភាព។ WAN protocols ពិពណ៌នាអំពី physical



(layer 1) និង data link (layer 2)។ តើ frame relay ប្រើអ្វីនៅលើ data link layer?

យើងប្រើ MAC addresses ពីព្រោះវាជា Ethernet ។ ប៉ុន្តែយើងត្រូវតែមានអ្វីមួយហៅថា **DLCI (Data Link Connection Identifier)**។

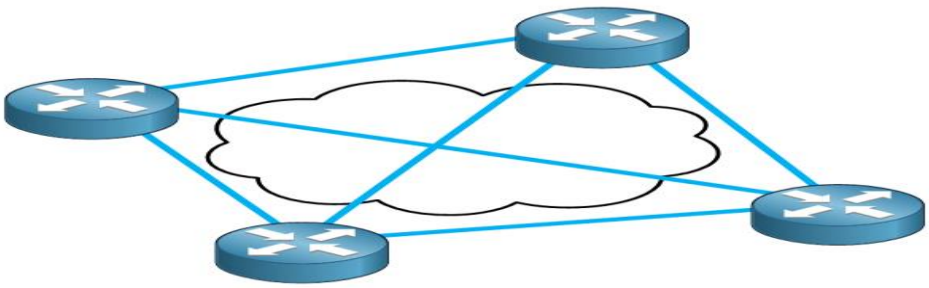
សម្រាប់ PVC នីមួយៗអ្នកនិងទទួលបាននូវ DLCI មួយសម្រាប់ Router មួយ។ នៅក្នុងឧទាហរណ៍នេះ យើងអាចយើញថាសម្រាប់ PVC រវាង Hub និង Spoke1។ យើងមាន DLCI 102 នៅលើ Hub និង DLCI 201 នៅលើ Spoke1។ រវាង Hub និង Spoke2 យើងមាន DLCI 103 នៅលើ Hub និង DLCI 301 នៅលើ Spoke2។ ចំពោះ DLCI គ្មានអ្វីក្រៅពីការសម្គាល់ឲ្យ data link layer សម្រាប់ PVC។

Router មិនស្គាល់ DLCI របស់ router នៅជ្រុងម្ខាងទៀតនោះទេ។ យើងនិងឃើញភាពខុសគ្នាបើអ្នកប្រៀបធៀបជាមួយ Ethernet។ ចំពោះ Ethernet អ្នកត្រូវតែដឹងពី MAC address នៃកុំព្យូទ័រនៅម្ខាងទៀតដើម្បីធ្វើការបញ្ជូនអ្វីមួយ។

Frame-relay គាំទ្រចំពោះ topology ជាច្រើនដូចជា:

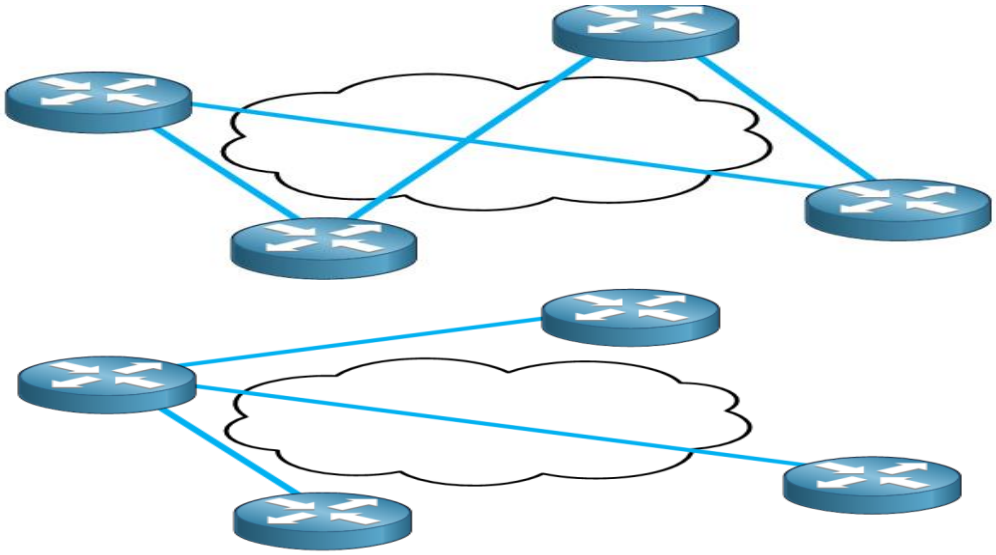
- Full mesh
- Partial Mesh
- Hub និង Spoke

សូមមើលការបង្ហាញពីឧទាហរណ៍នៃប្រភេទ topology នីមួយៗ:



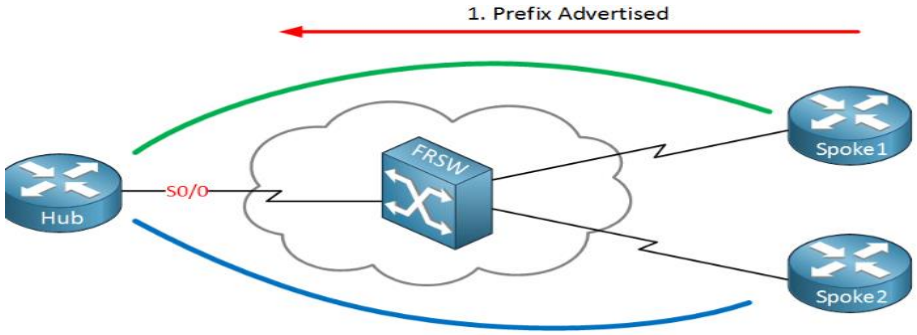
នេះគឺជា full-mesh topology។ អ្នកអាចឃើញថាមាន PVC មួយរវាងគ្រប់ router ។

នេះគឺជា partial-mesh។ ចំពោះ routers ដែលសំខាន់ត្រូវមានការភ្ជាប់ជាច្រើនមកកាន់ Routers ផ្សេងៗទៀត។ នេះគឺជា hub និង spoke model។ router នៅខាងឆ្វេងគឺជា hub ហើយ routers ផ្សេងទៀតគឺជា spokes។ បើ spokes ចង់ធ្វើការទំនាក់ទំនងជាមួយគ្នាទៅវិញទៅមក វាត្រូវបញ្ជូននូវចរាចរណ៍ទៅឲ្យ hub router។



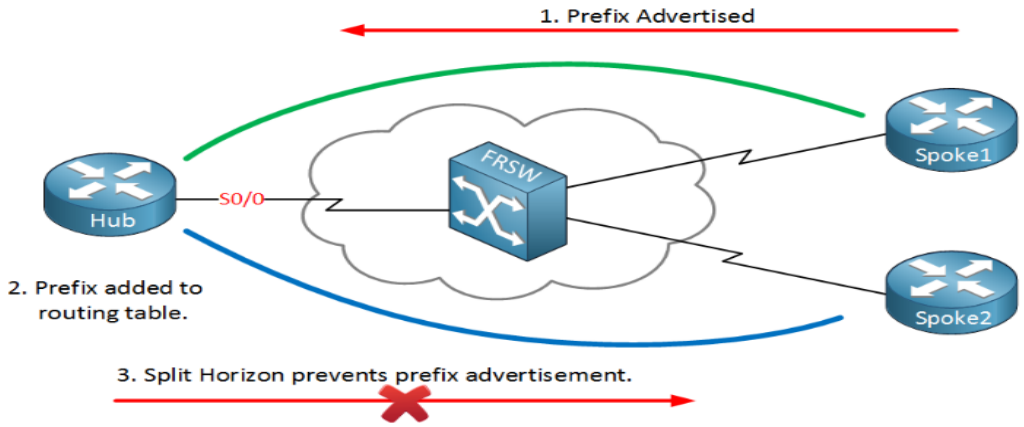
នេះមានន័យថា frame relay គឺជា multi-access ពីព្រោះថាគ្រប់ Routers អាច Access មកកាន់ ណេតវើកប៉ុន្តែ អ្នកមិនអាចបញ្ជូននូវ broadcasts តាមម frame relay ណេតវើកនោះទេ។ គ្មាន broadcast មានន័យថាអ្នកមិន អាចបញ្ជូននូវ multicast ចារចរណ៍ បាននោះទេ។ គ្មាន multicast មានន័យថាអ្នកស្ថិតនៅក្នុងបញ្ហានៃ routing protocols។ Rip version 2, OSPF និង EIGRP ប្រើ multicast។ តើមានន័យថាអ្នកមិនអាចប្រើ routing protocols ជាមួយ frame relay? មិនមែនអញ្ចឹងទេ វាគ្រាន់តែជាល្បិច:

- RIP, OSPF និង EIGRP អាចប្រើ unicast ជំនួសឲ្យ multicast
- មានវិធីសាស្ត្រមួយដើម្បីឲ្យអាចbroadcasts ទៅលើ frame relay ណេតវើកបាន



តើមានបញ្ហាអ្វីដែលយើងអាចជួបប្រទះជាមួយ frame relay និង routing? តើអ្នកចាំពីលក្ខណៈនៃ distance vector routing protocols (RIP and EIGRP) ដែរឬទេ?

នៅក្នុងរូបភាពខាងលើយើងបាន configure RIP នៅលើគ្រប់ routers ទាំងអស់។ Spoke1 កំពុងបញ្ជូននូវព័ត៌មានអំពី routing ឆ្ពោះទៅកាន់ Hub។ បើយើងពិនិត្យទៅលើ routing table យើងឃើញព័ត៌មានអំពី routing នៅលើ Hub។

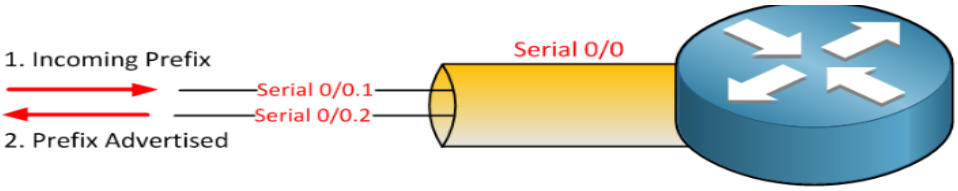


តើអ្នកបានចាំទេអំពី split-horizon? នៅពេលដែលអ្នករៀនអំពីអ្នកជិតខាងដែលមិនបានផ្សព្វផ្សាយត្រឡប់វិញ។ និយាយឲ្យចំគឺថាអ្វីដែលអ្នកបានរៀននៅលើ Interface ដែលអ្នកមិនបានផ្សព្វផ្សាយវាត្រឡប់មកលើ Interface ដដែលវិញ។

យើងកំពុងប្រើ PVC ពីរ ប៉ុន្តែនៅលើ Hub នៅតែមាន physical interface តែមួយគត់។ Split-horizon និងការពារនូវការផ្សព្វផ្សាយនៃព័ត៌មានអំពី routing មកកាន់ Spoke2។

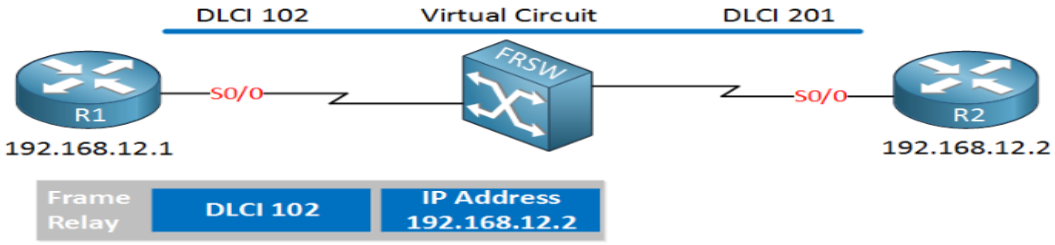
តើយើងដោះស្រាយបញ្ហាតាមរបៀបណា?

- អ្នកអាចបិទចោលនូវ split horizon (default នៅលើ physical interfaces)
- អ្នកអាចប្រើ sub-interfaces.



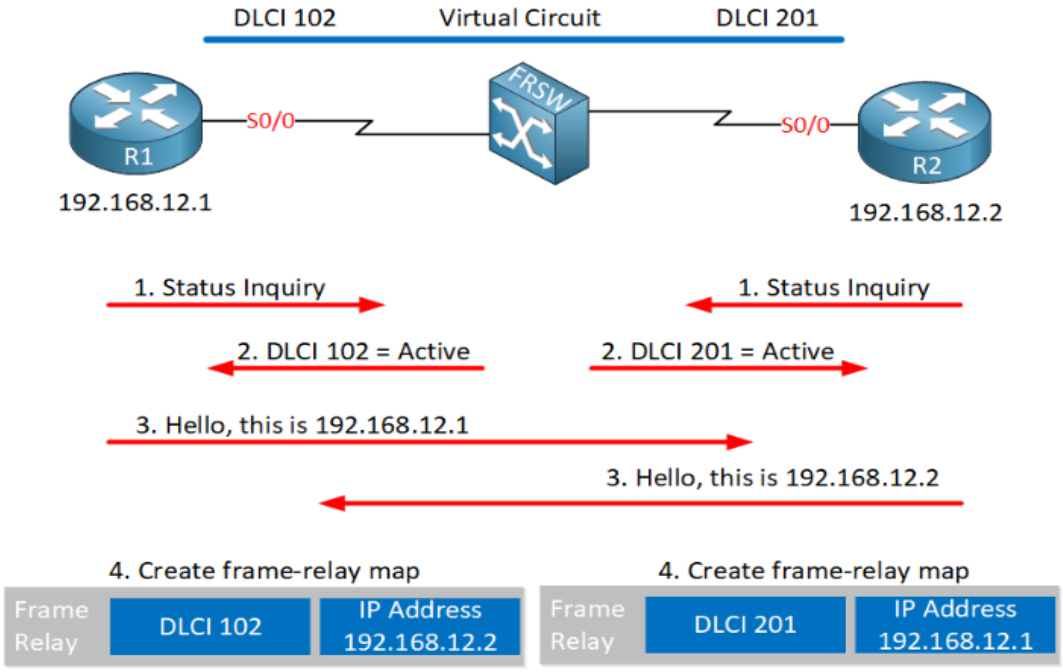
- បើអ្នកប្រើ sub-interface មួយនោះអ្នកមិនអាចមានបញ្ហា split-horizon នោះបានទេពីព្រោះថាអ្នកកំពុងរៀនអំពីព័ត៌មាននៃ routing នៅលើ serial0/0.1 និងការផ្សព្វផ្សាយព័តមានមកកាន់ serial0/0.2
- Frame-relay អាចប្រើ **point-to-point** sub-interfaces ឬ **point-to-multipoint** sub-interfaces។ បើអ្នកប្រើ point-to-point វានិងដោះស្រាយនូវបញ្ហានៃ split-horizon ប៉ុន្តែអ្នកនិងត្រូវការប្រើនូវ IP subnet ផ្សេងគ្នាសម្រាប់ PVC មួយ។ Point-to-multipoint មានន័យថាអ្នកមានបញ្ហា split-horizon ប៉ុន្តែអ្នកអាចប្រើ IP subnet តែមួយសម្រាប់គ្រប់ PVCs ទាំងអស់។
- តើអ្នកចាំ ARP ទេ? នៅពេលដែលយើងប្រើ ARP សម្រាប់ Ethernet យើងត្រូវការរៀនពី MAC address របស់កុំព្យូទ័រដែលយើងចង់បញ្ជូនអ្វីមួយទៅឲ្យ។ ARP កំនត់ destination IP address ឲ្យត្រូវជាមួយ destination MAC address
- Frame-relay ប្រើ **inverse ARP** ដែលវាខុសគ្នាបន្តិច។ អ្នកមិនបានដឹងពី DLCI របស់ Router ដែលស្ថិតនៅម្ខាងទៀតនោះទេ។ Inverse ARP កំពុងតែកំនត់ DLCI ឲ្យត្រូវជាមួយ IP address នៃម្ខាងទៀត

-



- R1 នៅក្នុងឧទាហរណ៍ខាងលើបានកំនត់ IP address នៃ R2 (192.16.12.2) ឲ្យត្រូវជាមួយ local DLCI 102 របស់វា។ នោះគេហៅថា inverse ARP។

Let's see it in more detail:



នៅពេលដែលយើង Configure ទៅលើ frame relay នៅទីនេះ។ តើមានអ្វីកើតមានឡើង?

ចំពោះ router របស់យើងនិងធ្វើការសាកសួរដោយប្រើ LMI

១-Frame relay Switch និងផ្តល់ឲ្យយើងនូវលេខ DLCI របស់យើង(ឬអ្នកអាច Configure វានៅពេលក្រោយ)

២-Routers បញ្ជូននូវ Hello message មួយជាមួយ IP addfress របស់វា

៣- routers ទទួលនូវ hello message ពីម្ខាងទៀតនិងបង្កើតការធ្វើឲ្យត្រូវគ្នាជាមួយ remote IP address + local DLCI number

R1 និងដឹងថាវាអាចភ្ជាប់ជាមួយ IP address 192.168.12.2 ដោយបញ្ជូននូវចរាចរណ៍តាម PVC ដែលមាន DLCI 102។ R2 និងដឹងថាវាអាចភ្ជាប់ IP address 192.168.12.1 តាមមរយៈ PVC មាន DLCI 201។

10-6-1-Cisco Frame-relay Switch Configuration

Frame-relay ត្រូវការនូវ frame-relay switch សម្រាប់ប្តូរពី DLCI មួយទៅកាន់មួយទៀតបង្កើតបានជា virtual circuit។ តាមធម្មតាអ្នកមិនត្រូវការគិតអំពី frame-relay switch ពីព្រោះថាវាជាអ្វីមួយដែលអ្នក ផ្តល់សេវាជាអ្នក Configure ។

បើអ្នកចង់ Configure ទៅលើ frame relay នៅក្នុងLab នោះអ្នកត្រូវការនូវ frame-relay switch។ សូមពិនិត្យមើលពីរូបភាព:

ក្នុងរូបភាពខាងលើ យើងមាន Routers ចំនួន៣ដែលត្រូវបង្កើត PVC រវាង R1 និង R2។ Router នៅកណ្តាលគឺជា frame-relay switch។

Configuration

សូមចាប់ផ្តើមជាមួយ frame-relay switch។ ដំហានដំបូងគឺត្រូវបើក frame-relay switching globally:

```
FRSWITCH(config)#frame-relay switching
```

Now we can focus on the interface configuration:

```
FRSWITCH(config)#interface serial1/1
FRSWITCH(config-if)#description R1
FRSWITCH(config-if)#clock rate 128000
```

ត្រូវដឹងថាអ្នកមាន clock rate ត្រូវបាន Configure នៅលើ DCE interfaces។ វាមិនទាក់ទងដោយផ្ទាល់ជាមួយ frame-relay នោះទេ ។ ប៉ុន្តែគេត្រូវការវាសម្រាប់ serial interfaces។ ឥឡូវនេះអ្នកអាច configure some frame-relay ជាមួយបញ្ជា:

```
FRSWITCH(config-if)#encapsulation frame-relay
FRSWITCH(config-if)#frame-relay intf-type dce
FRSWITCH(config-if)#frame-relay route 102 interface serial1/2 201
```

មានបញ្ជាចំនួន៣ដែលអ្នកត្រូវប្រើជាមួយ frame-relay:

- យើងត្រូវបើក Frame-relay encapsulation
- Interace ត្រូវតែ Configure ជា frame-relay DCE.
- យើងប្រាប់ interface ដើម្បីប្តូរអ្វីៗដែលមកដល់ជា DLCI 102 មកកាន់ interface S1/2 ជា DLCI 201 សូមពិនិត្យការ Configure ទៅលើ Interface ដែលភ្ជាប់មកកាន់ R2:

```
FRSWITCH(config)#interface serial1/2
FRSWITCH(config-if)#clock rate 128000
FRSWITCH(config-if)#description R2
FRSWITCH(config-if)#encapsulation frame-relay
FRSWITCH(config-if)#frame-relay intf-type dce
FRSWITCH(config-if)#frame-relay route 201 interface serial1/1 102
```

ការ Configgure ខាងលើគឺជាដូចគ្នា ប៉ុន្តែបញ្ជា frame-relay route command ស្ថិតក្នុងលក្ខណៈផ្ទុយគ្នា។

មានបញ្ហាតែមួយគត់ដែលត្រូវធ្វើគឺបើក interfaces នៃ R1 និង R2:

```
R1(config)#interface serial 0/0/0
R1(config-if)#encapsulation frame-relay
R2(config)#interface Serial 0/0/0
R2(config-if)#encapsulation frame-relay
```

អ្វីៗមានរួចជាស្រេច។ សូមពិនិត្យមើលថាតើ PVCs កំពុងដំណើរការឬទេ។ យើងអាចត្រួតពិនិត្យជាមួយបញ្ជាដូចខាងក្រោម:

```
FRSWITCH#show frame-relay route
Input Intf      Input Dlci      Output Intf     Output Dlci     Status
Serial1/1       102             Serial1/2       201             active
Serial1/2       201             Serial1/1       102             active
```

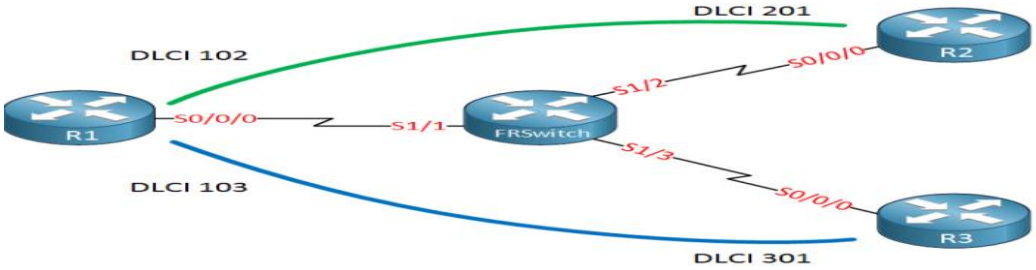
ខាងលើបញ្ជាក់ថា PVC គឺកំពុងតែមានសកម្មភាព។ ត្រូវផ្ទៀងផ្ទាត់ចំពោះ R1 និង R2:

```
R1#show frame-relay pvc

PVC Statistics for interface Serial0/0/0 (Frame Relay DTE)
```

```
            Active     Inactive      Deleted       Static
  Local         0           0           0           0
  Switched      0           0           0           0
  Unused        1           0           0           0

DLCI = 102, DLCI USAGE = UNUSED, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/0

  input pkts 0             output pkts 0           in bytes 0
  out bytes 0              dropped pkts 0          in pkts dropped 0
  out pkts dropped 0               out bytes dropped 0
  in FECN pkts 0           in BECN pkts 0          out FECN pkts 0
  out BECN pkts 0          in DE pkts 0            out DE pkts 0
  out bcast pkts 0         out bcast bytes 0
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  pvc create time 00:00:25, last time pvc status changed 00:00:25
R2#show frame-relay pvc

PVC Statistics for interface Serial0/0/0 (Frame Relay DTE)

            Active     Inactive      Deleted       Static
  Local         0           0           0           0
  Switched      0           0           0           0
  Unused        1           0           0           0

DLCI = 201, DLCI USAGE = UNUSED, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/0

  input pkts 0             output pkts 0           in bytes 0
  out bytes 0              dropped pkts 0          in pkts dropped 0
  out pkts dropped 0               out bytes dropped 0
  in FECN pkts 0           in BECN pkts 0          out FECN pkts 0
  out BECN pkts 0          in DE pkts 0            out DE pkts 0
  out bcast pkts 0         out bcast bytes 0
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
```

```
  pvc create time 00:00:38, last time pvc status changed 00:00:38
```

ជាការប្រសើរណាស់ដែល PVC កំពុងដំណើរការ។ frame-relay hub និង spoke topology មានប្រជាប្រិយភាព



ណាស់។ ត្រូវបន្ថែម router មួយឬច្រើនទៀតចូលទៅក្នុង network:

ខាងលើយើងបានបន្ថែម R3 និង PVC ទីពីររវាង R1 និង R3។ ត្រូវ Conffigure ទៅលើ frame-relay switch:

```
FRSWITCH(config)#interface Serial 1/1
FRSWITCH(config-if)#frame-relay route 103 interface Serial 1/3 301
```

យើងបាន Configure ទៅលើ interface ដែលភ្ជាប់មកកាន់ R1 ។ ដូច្នេះមានបញ្ហាមួយនៅសល់គឺត្រូវបន្ថែម frame-relay route មួយទៀត។ ត្រូវ configure S1/3 ដែលភ្ជាប់ជាមួយ R3:

```
FRSWITCH(config)#interface Serial 1/3
FRSWITCH(config-if)#description R3
FRSWITCH(config-if)#encapsulation frame-relay
FRSWITCH(config-if)#clock rate 128000
FRSWITCH(config-if)#frame-relay intf-type dce
FRSWITCH(config-if)#frame-relay route 301 interface Serial 1/1 103
```

Let’s enable frame-relay encapsulation on R3:

```
R3(config)#interface Serial 0/0/0
R3(config-if)#encapsulation frame-relay
```

And verify our work on the frame-relay switch:

```
FRSWITCH#show frame-relay route
Input Intf      Input Dlci      Output Intf     Output Dlci     Status
Serial1/1       102             Serial1/2       201             active
```

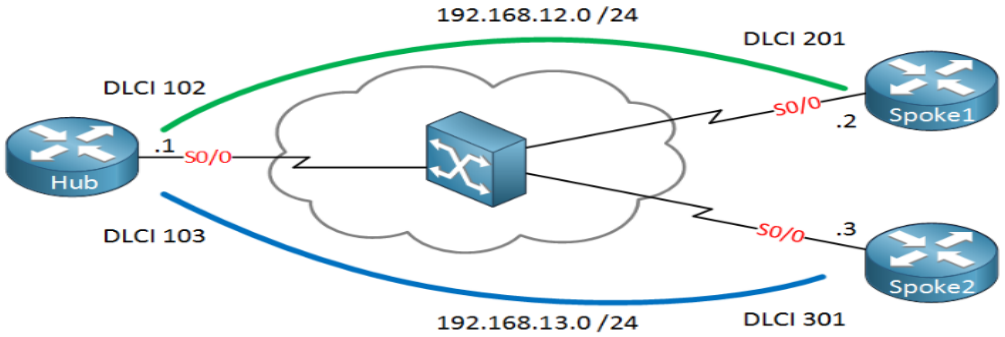
Serial1/1	103	Serial1/3	301	active
Serial1/2	201	Serial1/1	102	active
Serial1/3	301	Serial1/1	103	active

The new PVC is active, let's also check it on the routers:

```
R1#show frame-relay pvc | include PVC
PVC Statistics for interface Serial0/0/0 (Frame Relay DTE)
DLCI = 102, DLCI USAGE = UNUSED, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/0
DLCI = 103, DLCI USAGE = UNUSED, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/0
R2#show frame-relay pvc | include PVC
PVC Statistics for interface Serial0/0/0 (Frame Relay DTE)
DLCI = 201, DLCI USAGE = UNUSED, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/0
R3#show frame-relay pvc | include PVC
PVC Statistics for interface Serial0/0/0 (Frame Relay DTE)
DLCI = 301, DLCI USAGE = UNUSED, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/0
```

១០-៩-២-របៀប configure Frame-Relay Point-to-Point

នៅក្នុងមេរៀនមុនយើងបានដឹងអំពី frame-relay និងពន្យល់ពីរបៀប Configure ទៅលើ frame-relay point to multipoint។ នៅក្នុងមេរៀននេះយើងពិនិត្យទៅលើ frame-relay point-to-point ដែលមានលក្ខណៈងាយស្រួលក្នុងការ configure។ សូមពិនិត្យទៅលើ frame-relay point-to-point។ ខាងក្រោមនេះគឺជា topology ដែលយើងនិងប្រើវា:



យើងកំពុងប្រើ topology ដូចគ្នាដែលមានមួយទៀតខុសគ្នា។Point-to-point links ត្រូវការនូវ **IP Subnet per PVC**:

- Hub and Spoke1: 192.168.12.0 /24
- Hub and Spoke2: 192.168.13.0 /24

```
Hub(config)#interface serial 0/0
```

```
Hub(config-if)#encapsulation frame-relay
Hub(config)#interface serial 0/0.1 point-to-point
Hub(config-subif)#ip address 192.168.12.1 255.255.255.0
Hub(config-subif)#frame-relay interface-dlci 102
Hub(config)#interface serial 0/0.2 point-to-point
Hub(config-subif)#ip address 192.168.13.1 255.255.255.0
Hub(config-subif)#frame-relay interface-dlci 103
Spoke1(config)#interface s0/0
Spoke1(config-if)#encapsulation frame-relay
Spoke1(config-if)#exit
Spoke1(config)#interface serial 0/0.1 point-to-point
Spoke1(config-subif)#ip address 192.168.12.2 255.255.255.0
Spoke1(config-subif)#frame-relay interface-dlci 201
Spoke2(config)#interface serial 0/0
Spoke2(config-if)#encapsulation frame-relay
Spoke2(config-if)#exit
Spoke2(config)#interface serial 0/0.1 point-to-point
Spoke2(config-subif)#ip address 192.168.13.3 255.255.255.0
Spoke2(config-subif)#frame-relay interface-dlci 301
```

នេះគឺជា Configuration សម្រាប់ hub និង spoke routers។ អ្នកត្រូវតែកំនត់ឲ្យច្បាស់ពី encapsulation frame-relay នៅលើ physical interface។ បញ្ហាដែលនៅសល់គឺនៅលើ sub-interfaces។ router មិនអាចដឹងនិងរក sub-interfaces មួយណាដែលជារបស់ DLCI បានទេ។ ដូច្នេះអ្នកត្រូវតែ configure។ យើងនិងមិនប្រើបញ្ជា frame-relay map សម្រាប់ point-to-point sub-interfaces ។ ប៉ុន្តែអ្នកត្រូវតែប្រើបញ្ជា **frame-relay interface-dlci** ។ សូមពិនិត្យមើលការ Configure RIP

```
Hub(config)#router rip
Hub(config-router)#no auto-summary
Hub(config-router)#version 2
Hub(config-router)#network 192.168.12.0
Hub(config-router)#network 192.168.13.0
Spoke1(config)#interface loopback 0
Spoke1(config-if)#ip address 2.2.2.2 255.255.255.0
Spoke1(config)#router rip
Spoke1(config-router)#version 2
Spoke1(config-router)#no auto-summary
```

```
Spoke1(config-router)#network192.168.12.0
Spoke1(config-router)#network 2.0.0.0
Spoke2(config)#router rip
Spoke2(config-router)#version 2
Spoke2(config-router)#no auto-summary
Spoke2(config-router)#network 192.168.13.0
```

សូមពិនិត្យមលើទៅលើ routing tables:

```
Hub#show ip route rip
    2.0.0.0/24 is subnetted, 1 subnets
R       2.2.2.0 [120/1] via 192.168.12.2, 00:00:08, Serial0/0.102
```

Our Hub router has learned network 2.2.2.0 /24.

```
Spoke1#show ip route rip
R    192.168.13.0/24 [120/1] via 192.168.12.1, 00:00:08, Serial0/0.201
```

Spoke1 has learned about network 192.168.13.0 /24.

```
Spoke2#show ip route rip
R    192.168.12.0/24 [120/1] via 192.168.13.1, 00:00:10, Serial0/0.301
    2.0.0.0/24 is subnetted, 1 subnets
R       2.2.2.0 [120/2] via 192.168.13.1, 00:00:10, Serial0/0.301
```

ហើយ spoke2 បានដឹងពីណេតវើក192.168.12.0 /24 and 2.2.2.0 /24។ បើពិនិត្យឲ្យបានដិតដិតអ្នកអាចឃើញថា next hop IP address គឺ 192.168.13.1 (Hub router)។

១0-៩-៣-របៀប configure Frame-Relay Point-to-Multipoint

នេះគឺជាពេលវេលាដែលយើងត្រូវពិនិត្យទៅលើការ Configure ទៅលើ frame-relay point-to-multipoint។ បើអ្នកគ្មានគំនិតអំពី frame-relay ឬ PVC, DLCI ឬ LMI គឺត្រូវចាប់ផ្តើមជាមួយការណែនាំឲ្យស្គាលពី frame-relay ជាមុនសិន។

ខាងលើគឺជា topology ដែលយើងនិងប្រើវា។ មាន 3 routers នៅក្នុង hub និង spoke model។ មាន PVCs ពីរហើយអ្នកអាចឃើញពីលេខ DLCI នៅក្នុងរូបភាព។ យើងប្រើ subnet តែមួយគឺ (192.168.123.0 /24) ។ ដូច្នេះយើងនិងចាប់ផ្តើមជាមួយ frame-relay point-to-multipoint។ សូមរៀបចំទៅលើ interface:

```
Hub(config)#interface serial 0/0
```

```
Hub(config-if)#encapsulation frame-relay
Spoke1(config)#interface serial 0/0
Spoke1(config-if)#encapsulation frame-relay
Spoke2(config)#interface serial 0/0
Spoke2(config-if)#encapsulation frame-relay
```

យើងនិងប្តូរប្រភេទនៃ encapsulation មកជា frame-relay សម្រាប់គ្រប់ interfaces។ ត្រូវធ្វើងផ្ទាត់ឲ្យដឹងថា PVCs កំពុងដំណើរការ។

```
Hub#show frame-relay pvc
PVC Statistics for interface Serial0/0 (Frame Relay DTE)
            Active     Inactive      Deleted       Static
  Local         2            0            0            0
  Switched      0            0            0            0
  Unused        0            0            0            0

DLCI = 102, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/0

  input pkts 12           output pkts 11          in bytes 1108
  out bytes 1074          dropped pkts 0          in pkts dropped 0
  out pkts dropped 0              out bytes dropped 0
  in FECN pkts 0          in BECN pkts 0          out FECN pkts 0
  out BECN pkts 0         in DE pkts 0            out DE pkts 0
  out bcast pkts 1        out bcast bytes 34
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
  pvc create time 00:15:37, last time pvc status changed 00:15:37

DLCI = 103, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/0

  input pkts 12           output pkts 11          in bytes 1108
  out bytes 1074          dropped pkts 0          in pkts dropped 0
  out pkts dropped 0              out bytes dropped 0
  in FECN pkts 0          in BECN pkts 0          out FECN pkts 0
  out BECN pkts 0         in DE pkts 0            out DE pkts 0
```

```
out bcast pkts 1        out bcast bytes 34
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
pvc create time 00:15:41, last time pvc status changed 00:15:41
```

បញ្ជា **show frame-relay pvc** ប្រាប់យើងឲ្យដឹងថា PVCs កំពុងមានសកម្មភាព។ អ្នកក៏អាចឃើញពីលេខ DLCI ផងដែរ។ វាប្រាប់ឲ្យដឹងថា layer 2 នៃ frame-relay របស់យើងកំពុងដំណើរការ។ នៅក្នុងករណីដែលមានបញ្ហាជាគំនិតល្អត្រូវតែផ្ទៀងផ្ទាត់ទៅលើ LMI:

```
Hub#show frame-relay lmi

LMI Statistics for interface Serial0/0 (Frame Relay DTE) LMI TYPE = ANSI
  Invalid Unnumbered info 0          Invalid Prot Disc 0
  Invalid dummy Call Ref 0           Invalid Msg Type 0
  Invalid Status Message 0           Invalid Lock Shift 0
  Invalid Information ID 0           Invalid Report IE Len 0
  Invalid Report Request 0           Invalid Keep IE Len 0
  Num Status Enq. Sent 147           Num Status msgs Rcvd 148
  Num Update Status Rcvd 0           Num Status Timeouts 0
  Last Full Status Req 00:00:35              Last Full Status Rcvd 00:00:35
```

យើងប្រើបញ្ជា **show frame-relay lmi** ដើម្បីមើលព័ត៌មានអំពី LMI ។ វាប្រាប់យើងឲ្យដឹងថាយើងកំពុងប្រើប្រភេទនៃ ANSI។ វាមិនជាបញ្ហានោះទេដែលអ្នកប្រើមួយណានោះទេដរាបណាមានដូចគ្នានៅគ្រប់ Routers ។ ដោយសារតែ Layer 2 កំពុងដំណើរការ យើងនិង Configure នូវ IP addresses មួយចំនួនហើយពិនិត្យមើលថាតើ Layer 3 កំពុងដំណើរការដែរ:

```
Hub(config)#interface serial 0/0
Hub(config-if)#ip address 192.168.123.1
Spoke1(config)#interface serial 0/0
Spoke1(config-if)#ip address 192.168.123.2
Spoke2(config)#interface serial 0/0
Spoke2(config-if)#ip address 192.168.123.3
```

សូមពិនិត្យមើលយើងអាចភ្ជាប់ទៅកាន់ម្ខាងទៀត:

```
Hub#ping 192.168.123.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.2, timeout is 2 seconds:
```

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/8/24 ms
Hub#ping 192.168.123.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/8 ms
```

ដូចដែលអ្នកបានឃើញ hub router អាចភ្ជាប់មកខាង Spoke routers ទាំងពីរបាន។ នេះក៏ដោយសារតែ Inverse ARP ត្រូវបានបើកជា default។ យើងអាចត្រួតពិនិត្យទៅលើ frame-relay maps ដើម្បីបញ្ជាក់បន្ថែមពីបញ្ហានេះ:

```
Hub#show frame-relay map
Serial0/0 (up): ip 192.168.123.2 dlci 102(0x66,0x1860), dynamic,
        broadcast,, status defined, active
Serial0/0 (up): ip 192.168.123.3 dlci 103(0x67,0x1870), dynamic,
        broadcast,, status defined, active
Spoke1#show frame-relay map
Serial0/0 (up): ip 192.168.123.1 dlci 201(0xC9,0x3090), dynamic,
        broadcast,, status defined, active
Spoke2#show frame-relay map
Serial0/0 (up): ip 192.168.123.1 dlci 301(0x12D,0x48D0), dynamic,
        broadcast,, status defined, active
```

ដូចដែលអ្នកបានកំនត់ឲ្យត្រូវរវាង IP address និងលេខ DLCI។ មានចំណុចពីរដែលយើងត្រូវចាប់អារម្មណ៍នៅទីនេះ។ ពាក្យបច្ចេកទេស **dynamic** មានន័យថា entry ត្រូវបានរៀនពីព្រោះតែ inverse ARP។ ពាក្យថា **broadcast** មានន័យថាយើងបញ្ជូននូវ broadcast ឬ multicast ឆ្លងតាម PVC។

- Configurations
- Hub
- Spoke1
- Spoke2

តោះយើងបិទចំពោះ Inverse ARP ហើយបង្កើតនូវការកំនត់ឲ្យត្រូវគ្នាដោយយើងផ្ទាល់

```
Hub(config)#interface serial 0/0
```

```
Hub(config-if)#no frame-relay inverse-arp
Spoke1(config)#interface serial 0/0
Spoke1(config-if)#no frame-relay inverse-arp
Spoke2(config)#interface serial 0/0
Spoke2(config-if)#no frame-relay inverse-arp
```

ប្រើបញ្ជា **no frame-relay inverse-arp** ដើម្បីបិទវាចោល។

```
Hub#clear frame-relay inarp
Spoke1#clear frame-relay inarp
Spoke2#clear frame-relay inarp
```

And we'll use **clear frame-relay inarp** to get rid of the current frame-relay maps that were created using inverse ARP.

You'll see that connectivity is now impossible:

```
Hub#ping 192.168.123.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
Hub#ping 192.168.123.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.3, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

Router មិនបានដឹងពីកន្លែងណាដែលត្រូវផ្ញើនូវ IP packets របស់វាទៅនោះទេដោយសារតែយើងគ្មាន frame-relay maps។ ត្រូវបង្កើតវាដោយខ្លួនយើងផ្ទាល់:

```
Hub(config)#interface serial 0/0
Hub(config-if)#frame-relay map ip 192.168.123.2 102 broadcast
Hub(config-if)#frame-relay map ip 192.168.123.3 103 broadcast
Spoke1(config)#interface serial 0/0
Spoke1(config-if)#frame-relay map ip 192.168.123.1 201 broadcast
```

```
Spoke2(config)#interface serial 0/0
Spoke2(config-if)#frame-relay map ip 192.168.123.1 301 broadcast
```

ការប្រើបញ្ជា **frame-relay map** ដើម្បីកំនត់ IP address របស់អ្នកជិតខាងឲ្យត្រូវជាមួយលេខ DLCIផ្ទាល់ខ្លួនរបស់អ្នក។ ចំពោះពាក្យថា **broadcast** គឺជាក់ក៏បាន។ យើងត្រូវការវានៅលើ routers របស់យើងបើអ្នកចង់បញ្ជូនបន្តនូវ multicast ឬ broadcast ចារចរណ៍ (ឬបើអ្នកចង់ឲ្យដំណើរការនូវ routing protocol ដូចជា RIP,OSPF or EIGRP)នៅលើ frame-relay network។ Routing Protocols ដូចជា OSPF,EIGRP ឬ RIPv2 ប្រើ multicast ។ ដូច្នេះហើយអ្នកត្រូវប្រើពាក្យថា “broadcast” ។ វាក៏អាច Configure ចំពោះ Routing Protocol ទាំងនោះឲ្យប្រើ unicast ដោយប្រើបញ្ជា neighbour នៅក្នុង RIP,OSPF ឬ EIGRP configuration ។
សូមពិនិត្យមើលទៅលើការកំនត់ដូចខាងក្រោម:

```
Hub#show frame-relay map
Serial0/0 (up): ip 192.168.123.2 dlci 102(0x66,0x1860), static,
        broadcast,
        CISCO, status defined, active
Serial0/0 (up): ip 192.168.123.3 dlci 103(0x67,0x1870), static,
        broadcast,
        CISCO, status defined, active
```

ដូចអ្នកបានឃើញខាងលើ។ ពាក្យ static ប្រាប់អ្នកថាវាគឺជាការកំនត់ frame-relay ដែលបាន Configure ។ ដូច្នេះអ្នកអាច ping បាន

```
Hub#ping 192.168.123.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/8/24 ms
Hub#ping 192.168.123.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/8 ms
```

គ្មានបញ្ហាកើតមានឡើងនោះទេ។ ឥឡូវនេះអ្នកអាចឃើញពីរបៀបនៃការ Configure frame-relay point-to-multipoint, របៀបត្រួតពិនិត្យទៅលើ PVCs, LMI និងពីការកំនត់ frame-relay ជាមួយឬគ្មានInverse ARP

- Configurations
- Hub
- Spoke1
- Spoke2

យើងត្រូវពិនិត្យមើលដោយខ្លួនអ្នក។ អ្នកនិងអាចរកឃើញពីការ Configure នៃឧបករណ៍នីមួយៗ។ អ្នកអាចឃើញវាដោយប្រើ physical interfaces ប៉ុន្តែអ្នកក៏អាចប្រើ sub-interfaces បានដែរ។ សូមពិនិត្យមើលខាងក្រោម:

```
Hub(config)#default interface serial 0/0
Building configuration...
Interface Serial0/0 set to default configuration
Spoke1(config)#default interface serial 0/0
Building configuration...

Interface Serial0/0 set to default configuration
Spoke2(config)#default interface serial 0/0
Building configuration...
Interface Serial0/0 set to default configuration
```

បញ្ជា default interface ប្រើសម្រាប់ reset interface configuration របស់អ្នក។ ឥឡូវនេះត្រូវ Configure ទៅលើ interfaces:

```
Hub(config)#interface serial 0/0
Hub(config-if)#encapsulation frame-relay
Spoke1(config)#interface serial 0/0
Spoke1(config-if)#encapsulation frame-relay
Spoke2(config)#interface serial 0/0
Spoke2(config-if)#encapsulation frame-relay
```

នៅលើ physical interfaces យើងត្រូវតែ Configure គឺប្រើ frame-relay។

Now I can create sub-interfaces:

```
Hub(config)#interface serial 0/0.123 multipoint
Hub(config-subif)#ip address 192.168.123.1 255.255.255.0
Hub(config-subif)#frame-relay map ip 192.168.123.2 102 broadcast
Hub(config-subif)#frame-relay map ip 192.168.123.3 103 broadcast
```

```
Spoke1(config)#interface serial 0/0.201 multipoint
Spoke1(config-subif)#ip address 192.168.123.2 255.255.255.0
Spoke1(config-subif)#frame-relay map ip 192.168.123.1 201 broadcast
Spoke2(config)#interface serial 0/0.301 multipoint
Spoke2(config-subif)#ip address 192.168.123.3 255.255.255.0
Spoke2(config-subif)#frame-relay map ip 192.168.123.1 301 broadcast
```

អ្នកអាចប្រើលេខ sub-interface ណាមួយដែលអ្នកចង់ប្រើ។ យើងប្រើលេខ DLCI សម្រាប់លេខ sub-interface ប៉ុន្តែអ្នកមិនអាចធ្វើវានៅលើ hub router ពីព្រោះថាវាមានលេខ DLCI ចំនួនពីរ។ គ្មានវិធីណាមួយសម្រាប់របស់ router ដែលប្រាប់ថាលេខ DLCI មួយណាដែលជារបស់ sub-interface មួយណានោះទេ។ ដូច្នេះយើងត្រូវការប្រើបញ្ជា frame-relay map ។ តើ Configuration របស់យើងដំណើរការដែរឬទេ? សូមធ្វើការ ping

```
Hub#ping 192.168.123.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/8/24 ms
Hub#ping 192.168.123.3
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 192.168.123.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/4/8 ms
```

ជាការប្រសើរណាស់ វាដំណើរការ។

ជំហានបន្ទាប់គឺត្រូវ Configure ចំពោះ distance vector routing protocol ។ ដូច្នេះយើងអាចបង្ហាញពីរបៀបធ្វើការជាមួយ split-horizon។ យើងនិងប្រើ RIP ប៉ុន្តែអ្នកក៏អាចប្រើ EIGRP:

```
Hub(config)#router rip
Hub(config-router)#no auto-summary
Hub(config-router)#version 2
Hub(config-router)#ណេតវើក192.168.123.0
Spoke1(config)#interface loopback 0
Spoke1(config-if)#ip address 2.2.2.2 255.255.255.0
```

```
Spoke1(config)#router rip
Spoke1(config-router)#version 2
Spoke1(config-router)#no auto-summary
Spoke1(config-router)#network 192.168.123.0
Spoke1(config-router)#network 2.0.0.0
Spoke2(config)#router rip
Spoke2(config-router)#version 2
Spoke2(config-router)#no auto-summary
Spoke2(config-router)#network 192.168.123.0
```

ដូចដែលអ្នកបានឃើញហើយយើងបានបើក RIP version 2 និងបង្កើតនូវ loopback interface នៅលើ router Spoke1 ។ ដូច្នេះយើងមានអ្វីដែលត្រូវផ្សព្វផ្សាយ។ សូមពិនិត្យមើលពី routing tables:

```
Hub#show ip route rip
     2.0.0.0/24 is subnetted, 1 subnets
R       2.2.2.0 [120/1] via 192.168.123.2, 00:00:09, Serial0/0.123
```

hub router បានរៀនពីណេតវើក2.2.2.0 /24 ពី Spoke1 ។ តើ Spoke2 យ៉ាងណាដែរ?

```
Spoke2#show ip route rip
```

គ្មានអ្វីទាំងអស់នៅទីនោះ។ នេះពីព្រោះថា split horizon នៅលើ Hub router ត្រូវបានបិទចោលនូវការផ្សព្វផ្សាយ។ ត្រូវដោះស្រាយបញ្ហាដូចខាងក្រោម:

```
Hub(config)#interface serial 0/0.123
Hub(config-subif)#no ip split-horizon ?
  eigrp  Enhanced Interior Gateway Routing Protocol (EIGRP)
  <cr>
Hub(config-subif)#no ip split-horizon
```

នៅពេលដែលយើងប្រើ RIP អ្នកអាចប្រើ no ip split-horizon ដើម្បីបិទវាចោលបាន។ មានបញ្ហាដាច់ដោយឡែកសម្រាប់ EIGRP ដូចអ្នកបានឃើញខាងលើ។ សូមត្រួតពិនិត្យទៅលើ spoke2 ម្តងទៀត

```
Spoke2#show ip route rip
     2.0.0.0/24 is subnetted, 1 subnets
R       2.2.2.0 [120/2] via 192.168.123.2, 00:00:03, Serial0/0.301
```

បញ្ហាត្រូវបានដោះស្រាយ។ យើងអាចឃើញងអ្វីដែលមាននៅក្នុង routing table។ តើវាអាចភ្ជាប់គ្នាបានឬទេ?

```
Spoke2#ping 2.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

ទោះបីជាវាស្ថិតនៅក្នុង routing table ក៏ដោយក៏វាមិនអាច ping បានដែរ។ វាកើតឡើងពីព្រោះតែ next hop IP address (192.168.123.2) មិនអាចភ្ជាបមកកាន់ Spoke2 បាននោះទេ។ យើងបង្កើតតែ frame-relay map មួយដើម្បីភ្ជាបមកកាន់ Hub router មិនមែន spoke1 នោះទេ។ សូមបង្កើត mappings ពីរបន្ថែមទៀត។ ដូច្នេះ spoke1 និង spoke2 អាចភ្ជាប់គ្នាទៅវិញទៅមក:

```
Spoke1(config)#interface serial 0/0.201
Spoke1(config-subif)#frame-relay map ip 192.168.123.3 201
Spoke2(config)#interface serial 0/0.301
Spoke2(config-subif)#frame-relay map ip 192.168.123.2 301
```

ចំពោះ frame-relay maps ខាងលើនិងប្រាកដថា spoke routers អាចស្គាល់គ្នាទៅវិញទៅមកបាន។ សូមព្យាយាមខាងក្រោម:

```
Spoke2#ping 2.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/9/20 ms
```

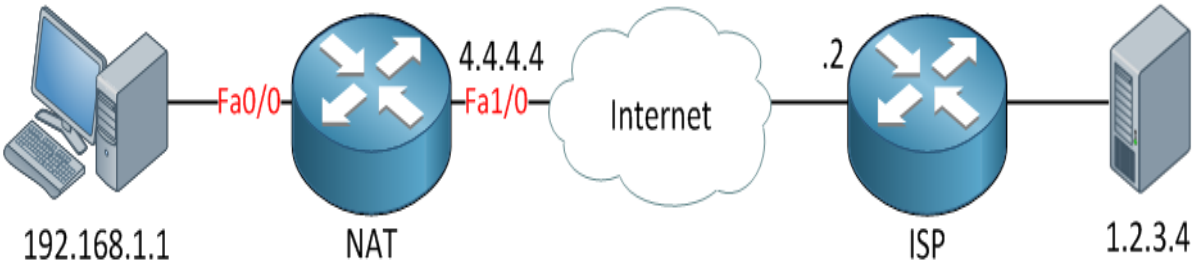
វាមានគ្រប់អស់ហើយ។ អ្នកអាចដឹងពីរបៀបនៃការ configure frame-relay point-to-multipoint នៅលើ physical និង sub-interfaces។ អ្នកក៏អាចឃើញពីរបៀបនៃ split horizon ដែលអាចបណ្តាលឲ្យមានបញ្ហាកើតឡើងនិងពីរបៀបនៃការដោះស្រាយបញ្ហាដោយបិទ split horizon នោះចោល។

ជំពូកទី១១

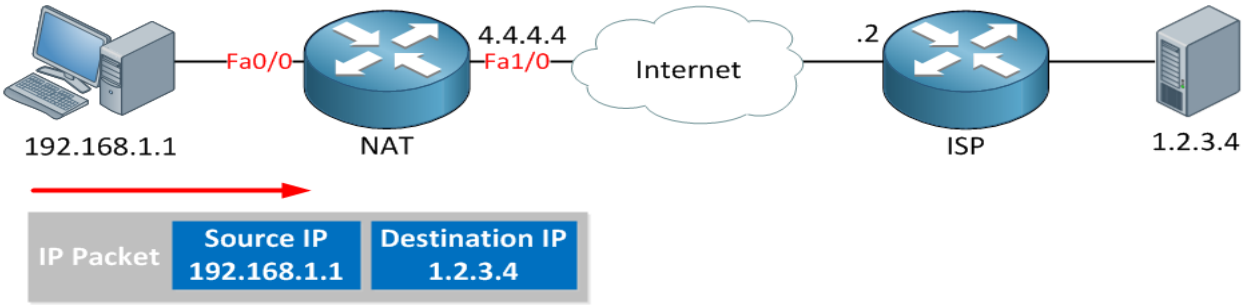
NAT and PAT

១១-១-សេចក្តីផ្តើមចំពោះ NAT និង PAT

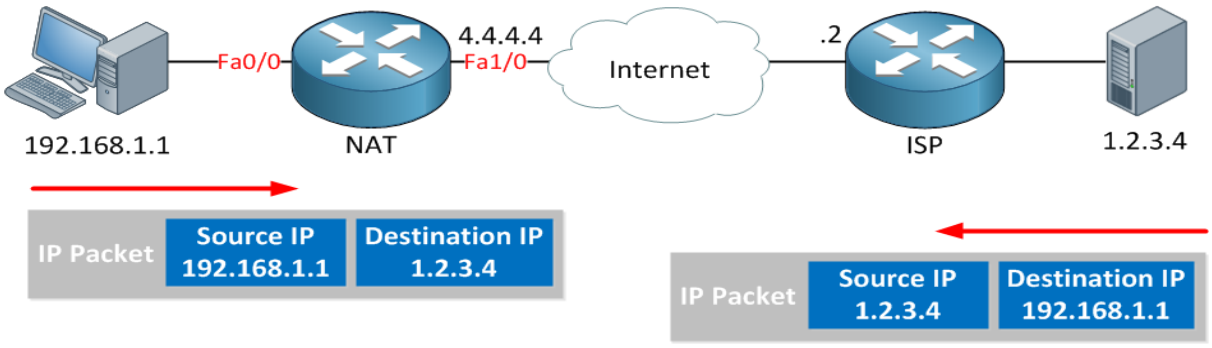
បើគ្មាន NAT ឬ PAT នោះទេ យើងយើងប្រហែលជាមិនអាចចូលប្រើប្រព័ន្ធ Internet បាននោះទេនៅក្នុងផ្ទះរបស់អ្នកដែលមានកុំព្យូទ័រច្រើន។ យើងនិងសិក្សាទៅលើ Topology ដូចខាងក្រោម:



នៅខាងឆ្វេងយើងមានកុំព្យូទ័រមួយនៅលើ LAN មួយដែលមាន IP address 192.168.1.1 បានភ្ជាប់ទៅនិង Router មួយ។ ពីក្រុមហ៊ុន ISP យើងទទួលបាន IP address 4.4.4.4 ហើយមាន Server មួយនៅលើប្រព័ន្ធ Internet ដែលកំពុងប្រើ IP address 1.2.3.4។ បើកុំព្យូទ័ររបស់យើងកំពុងបញ្ជូនអ្វីមួយមកកាន់ Server នោះអ្វីដែលមាននៅក្នុង IP Packet គឺ IP source និង destination IP address។



source IP address គឺជាកុំព្យូទ័ររបស់យើងហើយ destination IP address គឺជា Server ដែលអ្នកឃើញនៅក្នុង IP Packet ដែលបានបង្ហាញខាងលើ។

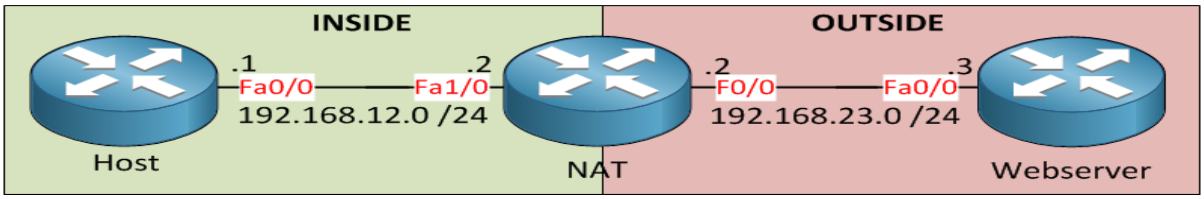


នៅពេលដែល Server របស់យើងបានឆ្លើយតបភ្លាម វាបង្កើតនូវ IP packet ដែលបានកំនត់ពី IP address របស់កុំព្យូរដែលជាគោលដៅហើយ Source IP address គឺជា IP address ផ្ទាល់ខ្លួនរបស់វា។

IP address របស់កុំព្យូទ័រនិង IP address នៅលើ Router គឺជា Private IP addresses ។ Private IP addresses ត្រូវបានប្រើសម្រាប់ LANs ហើយ public IP addresses ត្រូវបានប្រើសម្រាប់ Internet ។ នៅពេលនេះយើងនឹងចាប់ផ្តើម Configure NAT ហើយយើងនិងឃើញពីភាពខុសគ្នា ។

១១-២-របៀប Configure Static NAT នៅលើ Cisco IOS Router

សូមពិនិត្យមើលពីរបៀប configure static NAT នៅលើ Cisco router ។



ដូចរូបខាងលើមាន Routers ចំនួន៣ដែលមានឈ្មោះថា Host, NAT និង Webserver ។ ត្រូវគិតថា Host របស់យើងនៅក្នុង LAN ហើយ Webserver ស្ថិតនៅក្នុង Internet ។ NAT router នៅកណ្តាលដែលភ្ជាប់ទៅនិង Internet ។

មានវិធីសាស្ត្រមួយដែលគេប្រើនៅលើ Routers ។ យើងអាចបិទ Routing នៅលើ Router ដើម្បីឲ្យវាក្លាយជា Host ធម្មតាវិញដែលត្រូវការនូវ Default gateway មួយ ។

Host(config)#no ip routing

Webserver(config)#no ip routing

គេប្រើបញ្ជា no ip routing ដើម្បីបិទចោលនូវសមត្ថភាព Routing ។ ដូច្នេះ Routing tableត្រូវបានបាត់បង់ ។

Host#show ip route

Default gateway is not set

Host Gateway Last Use Total Uses Interface

ICMP redirect cache is empty

Webserver#show ip route

Default gateway is not set

Host Gateway Last Use Total Uses Interface

ICMP redirect cache is empty

ដូចអ្នកបានឃើញថាស្រាប់ Routing Table បានបាត់ ។ យើងនិង configure a default gateway នៅលើ Router Host និង Webserver ឬវាមោនអាចទាក់ទងគ្នាបាន ។

Host(config)#ip default-gateway 192.168.12.2

Webserver(config)#ip default-gateway 192.168.23.2

Routers ទាំងពីរប្រើ Router NAT ជា Default gateway របស់វា។ សូមសាកល្បងថាតើវាអាចស្គាល់គ្នាដែរឬទេ ?

Host#ping 192.168.23.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/12 ms

Reachability is no issue as you can see. Now let me show you a neat trick:

Webserver#debug ip packet

IP packet debugging is on

យើងប្រើបញ្ជា debug ip packet ដើម្បីមើលពី IP packets ដែលទទួលបាន។

Webserver#

IP: s=192.168.12.1 (FastEthernet0/0), d=192.168.23.3, len 100, rcvd 1

ដូចយើញខាងលើ Router របស់យើងបានទទួលនូវ IP packet មួយដែលមាន source IP address 192.168.12.1 និង destination IP address 192.168.23.3 ។

IP: tableid=0, s=192.168.23.3 (local), d=192.168.12.1 (FastEthernet0/0), routed via RIB

ហើយវានឹងឆ្លើយតបជាមួយ IP packet ដែលមាន source address 192.168.23.3 និង destination address 192.168.12.1 ។

ឥឡូវនេះ configure NAT អ្នកនឹងឃើញពីភាពខុសគ្នា:

NAT(config)#interface fastEthernet 1/0

NAT(config-if)#ip nat inside

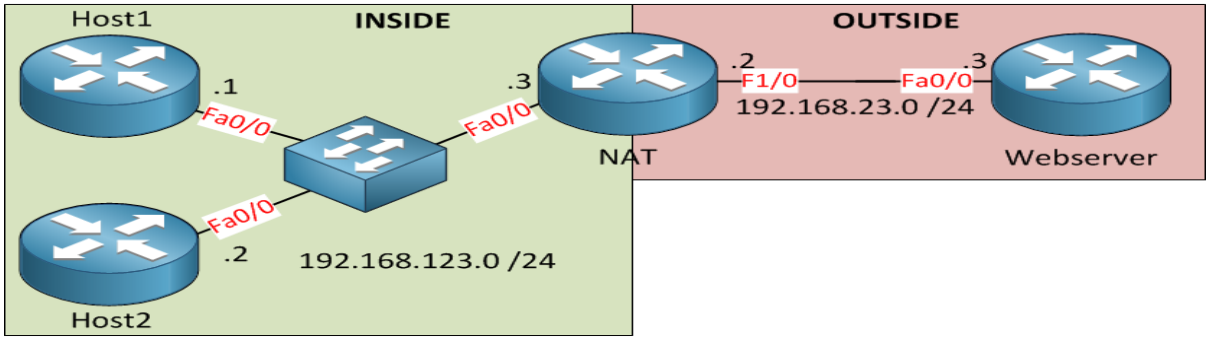
NAT(config)#interface fastEthernet 0/0

NAT(config-if)#ip nat outside

ជាដំបូងយើងត្រូវ Configure ចំពោះ inside និង outside interface ។ Host របស់យើងគឺខាង LAN ដូច្នេះវាជា Inside ។ Webserver របស់យើងគឺនៅលើ Internet ។ ដូច្នេះវាស្ថិតនៅខាងក្រៅ ណេតវើករបស់យើង។ ឥឡូវនេះយើងអាច Configure static NAT rule:

១១-៣-របៀប configure Dynamic NAT នៅលើ Cisco IOS Router

នេះជាពេលដែលត្រូវ dynamic NAT ដែលយើងប្រើបណ្តុំនៃ IP addresses សម្រាប់បកប្រែ។ យើងនិងប្រើនូវ Topology ដែលមាន Host ចំនួនពីរនិង Router ចំនួន១ដែលត្រូវសម្តែង NAT ។



នៅពេលនេះយើងមាន Host ចំនួន២ដែលជា Routers នៅខាងឆ្វេងហើយយើងប្រើ Subnet មួយផ្សេងទៀត។

Host1(config)#no ip routing

Host1(config)#ip default gateway 192.168.123.3

Host2(config)#no ip routing

Host2(config)#ip default-gateway 192.168.123.3

ជំហានបន្ទាប់ត្រូវ configure NAT:

NAT(config)#interface fastEthernet 0/0

NAT(config-if)#ip nat inside

NAT(config)#interface fastEthernet 1/0

NAT(config-if)#ip nat outside

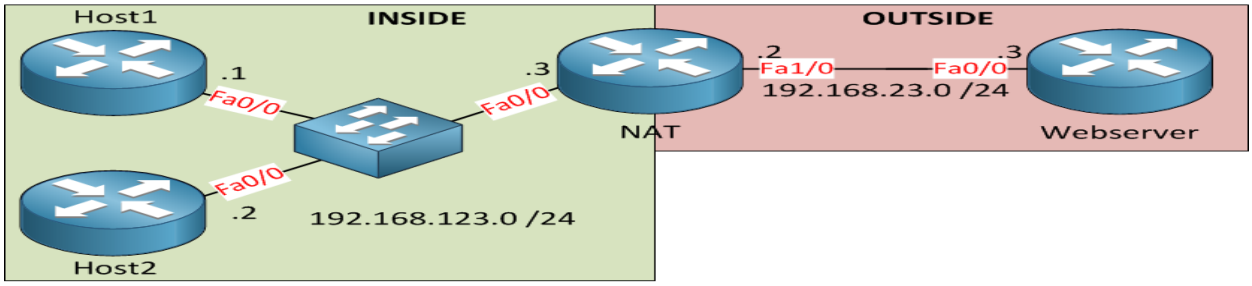
ជាដំបូងយើងត្រូវ Configure ឲ្យបានត្រឹមត្រូវចំពោះ inside និង Outside Interface ។ ឥឡូវនេះយើងបង្កើតនូវបណ្តុំនៃ IP addresses ដែលយើងអាចប្រើសម្រាប់បកប្រែ។

NAT(config)#ip nat pool MYPOOL 192.168.23.10 192.168.23.20 netmask 255.255.255.0

បញ្ជា ip nat pool អាចឲ្យយើងបង្កើតបានជាបណ្តុំនៃ IP addresses ដែលមានឈ្មោះថា “MYPOOL” ហើយមាន IP address 192.168.23.10 ដល់ 192.168.23.20។ យើងអាចជ្រើសរើសយក Hosts ដែលយើងចង់បកប្រែ។

របៀប Configure PAT នៅលើ Cisco IOS Router

យើងបានដឹងពីការ Configure នៃ static NAT និង dynamic NAT ។ ឥឡូវនេះត្រូវ Configure PAT ។ នេះគឺជា Topology ដែលយើងនិងប្រើវា:



ឥឡូវនេះរៀបចំ Host ដើម្បីប្រើជាមួយបញ្ជា “ip routing” ដើម្បីបិទវាឲ្យក្លាយជា host ដែលមិនឆ្លាត ។

```
Host1(config)#no ip routing

Host1(config)#ip default gateway 192.168.123.3

Host2(config)#no ip routing

Host2(config)#ip default-gateway 192.168.123.3
```

ជំហានបន្ទាប់គឺត្រូវ configure NAT:

```
NAT(config)#interface fastEthernet 0/0

NAT(config-if)#ip nat inside

NAT(config)#interface fastEthernet 1/0

NAT(config-if)#ip nat outside
```

លើសពីនេះទៅទៀត យើងត្រូវបង្កើតនូវ access-list ឲ្យត្រូវជាមួយ Hosts ទាំងពីរ:

```
NAT(config)#access-list 1 permit 192.168.123.0 0.0.0.255
```

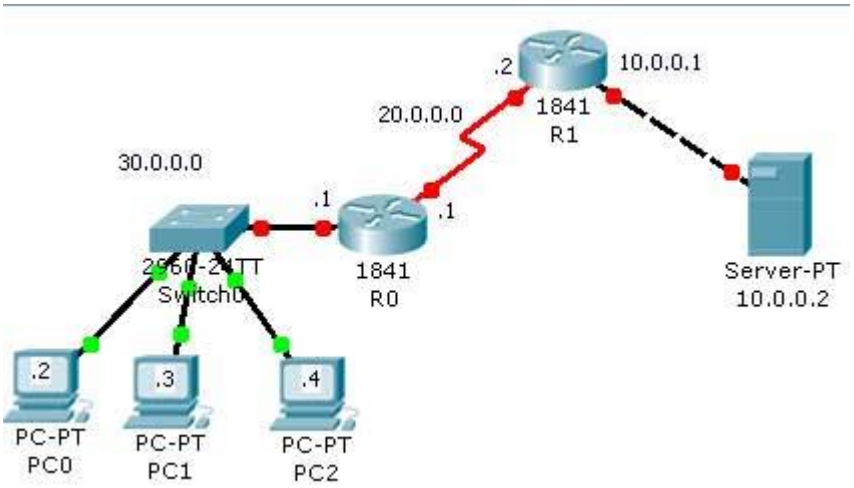
ហើយនូវជំហានចុងក្រោយត្រូវ configure PAT:

```
NAT(config)#ip nat inside source list 1 interface fastEthernet 1/0 overload
```

យើងបានប្រើ access-list 1 ជា inside source ហើយនិងបកប្រែវាឲ្យត្រូវជាមួយ IP address នៅលើ FastEthernet 1/0 ។

ឧទាហរណ៍នៃការ Configure static NAT និង Dynamic NAT

ការ Configure ចំពោះ Static NAT គឺជាវិធីមួយងាយស្រួល។ នៅក្នុងឧទាហរណ៍យើងមាន Web server មួយភ្ជាប់ជាមួយ Router មួយ។ Web server របស់យើងកំពុងប្រើ IP address 10.0.0.2 ។ នៅក្នុងក្រុមហ៊ុនចង់ប្រើ IP address:50.0.0.1 សម្រាប់ Server នេះ។ កិច្ចការរបស់យើងគឺត្រូវ Configure NAT នៅលើ Router ដែលត្រូវបកប្រែ 10.0.0.2 [ជា local web server ខាងក្នុង] ឲ្យត្រូវជាមួយ 50.0.0.1 [inside global ip address] ។



នៅលើ Router 1

Router>enable

Router#configure terminal

Router(config)#hostname R1

R1(config)#interface fastethernet 0/0

R1(config-if)#ip address 10.0.0.1 255.0.0.0

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#interface serial 0/0/0

R1(config-if)#ip address 20.0.0.2 255.0.0.0

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#ip route 30.0.0.0 255.0.0.0 20.0.0.1

R1(config)#ip nat inside source static 10.0.0.2 50.0.0.1

R1(config)#interface fastEthernet 0/0

R1(config-if)#ip nat inside

R1(config-if)#exit

R1(config)#interface serial 0/0/0

R1(config-if)#ip nat outside

R1(config-if)#exit

R1(config)#

នៅលើ R0

Router>enable

Router#configure terminal

Router(config)#hostname R0

R0(config)#interface fastethernet 0/0

R0(config-if)#ip address 30.0.0.1 255.0.0.0

R0(config-if)#no shutdown

R0(config-if)#exit

R0(config)#interface serial 0/0/0

R0(config-if)#ip address 20.0.0.1 255.0.0.0

R0(config-if)#clock rate 64000

R0(config-if)#bandwidth 64

R0(config-if)#no shutdown

R0(config-if)#exit

R0(config)#ip route 50.0.0.0 255.0.0.0 20.0.0.2

R0(config)#

ដូចអ្នកបានឃើញនៅក្នុង Configuration គ្មានការ route ដោយផ្ទាល់សម្រាប់ 10.0.0.2។ ដូច្នេះ PC ពីណេតវើក 30.0.0.0 និងមិនដឹងពីវានោះទេ។ វានិងចូលប្រើ 50.0.0.1 ដែលជា web server IP។ ដើម្បីតេស្តត្រូវ ping ពី 50.0.0.1 នោះនឹងទទួលបានលទ្ធផល។

Packet Tracer PC Command Line 1.0

PC>ping 50.0.0.1

Pinging 50.0.0.1 with 32 bytes of data:

Reply from 50.0.0.1: bytes=32 time=141ms TTL=126

Reply from 50.0.0.1: bytes=32 time=80ms TTL=126

Reply from 50.0.0.1: bytes=32 time=109ms TTL=126

Reply from 50.0.0.1: bytes=32 time=125ms TTL=126

Ping statistics for 50.0.0.1:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 80ms, Maximum = 141ms, Average = 113ms

ឥឡូវនេះ ping ពី 10.0.0.2 នោះទទួលបាន destination host unreachable error.

PC>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:

Reply from 30.0.0.1: Destination host unreachable.

Reply from 30.0.0.1: Destination host unreachable.

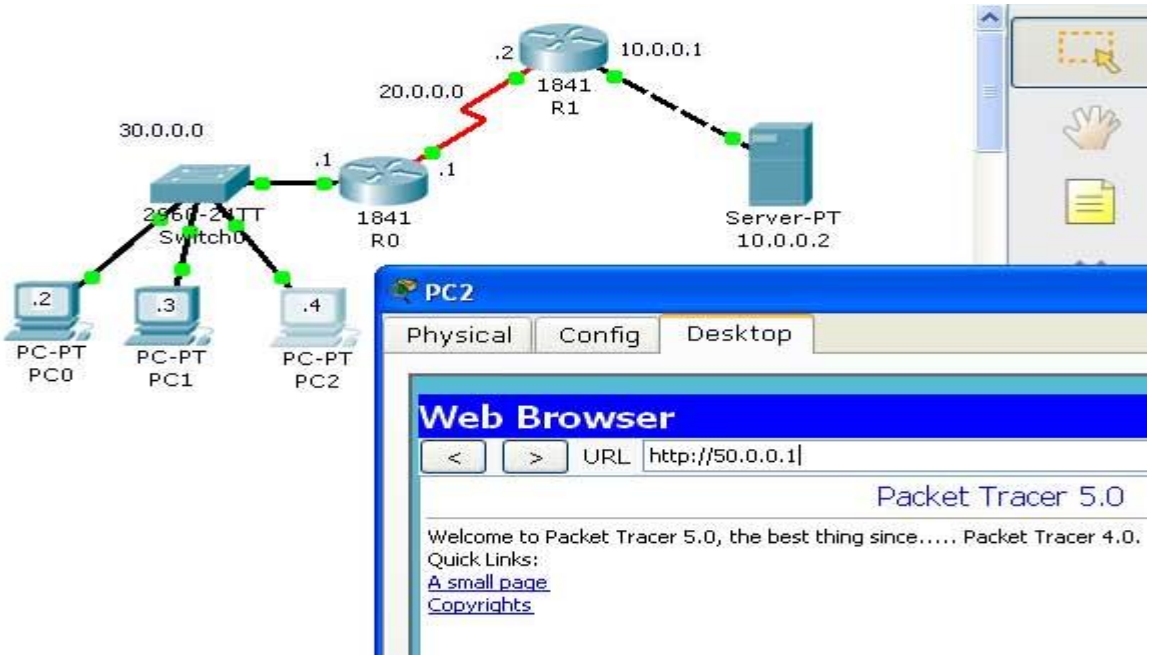
Reply from 30.0.0.1: Destination host unreachable.

Reply from 30.0.0.1: Destination host unreachable.

Ping statistics for 10.0.0.2:

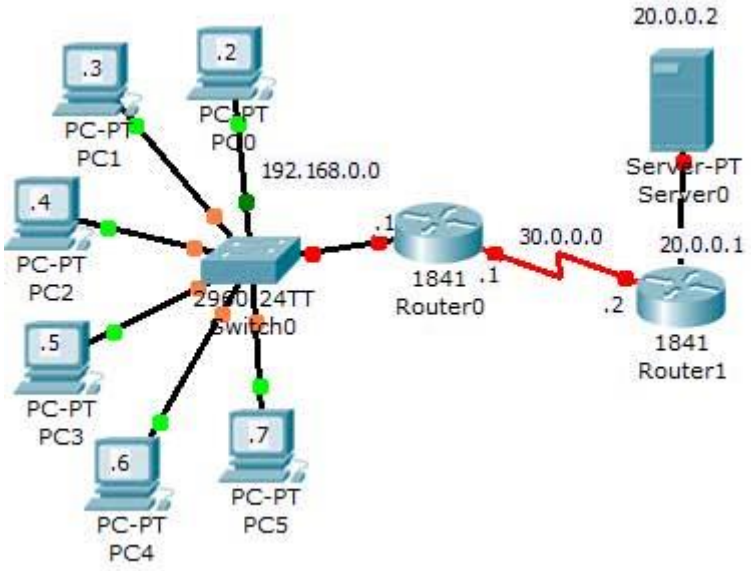
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss)

នេះជាការបង្ហាញពីរបៀបដែលក្រុមហ៊ុនប្រើ NAT ដើម្បីលាក់ IP addresses ខាងក្នុងរបស់គេពីខាងក្រៅ។ ឥឡូវនេះបើក Web browser ពី PC នៅក្នុង ណេតវើក30.0.0.0 និង browse ទៅកាន់ 50.0.0.1 site ។



ការ Configure ចំពោះ Dynamic NAT

ជាមួយ Dynamic NAT អ្នកត្រូវតែកំនត់ពីសំនុំនៃ IP addresses នៅលើឧបករណ៍បកប្រែ Address របស់អ្នក។ មួយគឺកំនត់ Address ខាងក្នុងដែលត្រូវអនុញ្ញាតឲ្យបកប្រែ(local address)និងកំនត់ពី Address ដែលត្រូវបកប្រែពីឲ្យត្រូវជាមួយ Address ខាងក្រៅ។



នៅក្នុងឧទាហរណ៍ ណេតវើកខាងក្នុងរបស់យើងគឺ 192.168.0.0។ យើងមាន public IP addresses ចំនួន ៥ គឺ 50.0.0.1 ដល់ 50.0.0.5 ។ Router1(1841 Router0) នឹងដើរតួនាទីជា NAT device ។

```
Router>enable

Router#configure terminal

Router(config)#hostname R1

R1(config)#interface fastethernet 0/0

R1(config-if)#ip address 192.168.0.1 255.0.0.0

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#interface serial 0/0/0

R1(config-if)#ip address 30.0.0.1 255.0.0.0

R1(config-if)#clock rate 64000

R1(config-if)#bandwidth 64

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#ip route 0.0.0.0 0.0.0.0 serial 0/0/0

R1(config)#access-list 1 permit 192.168.0.0 0.0.0.255

R1(config)#ip nat pool test 50.0.0.1 50.0.0.5 netmask 255.0.0.0

R1(config)#ip nat inside source list 1 pool test

R1(config)#interface fastEthernet 0/0

R1(config-if)#ip nat inside

R1(config-if)#exit

R1(config)#interface serial 0/0/0

R1(config-if)#ip nat outside

R1(config-if)#exit

R1(config)#exit
```

នៅលើ R2

Router>enable

Router#configure terminal

Router(config)#interface fastEthernet 0/0

Router(config-if)#ip address 20.0.0.1 255.0.0.0

Router(config-if)#no shutdown

Router(config-if)#exit

Router(config)#interface serial 0/0/0

Router(config-if)#ip address 30.0.0.2 255.0.0.0

Router(config-if)#no shutdown

Router(config-if)#exit

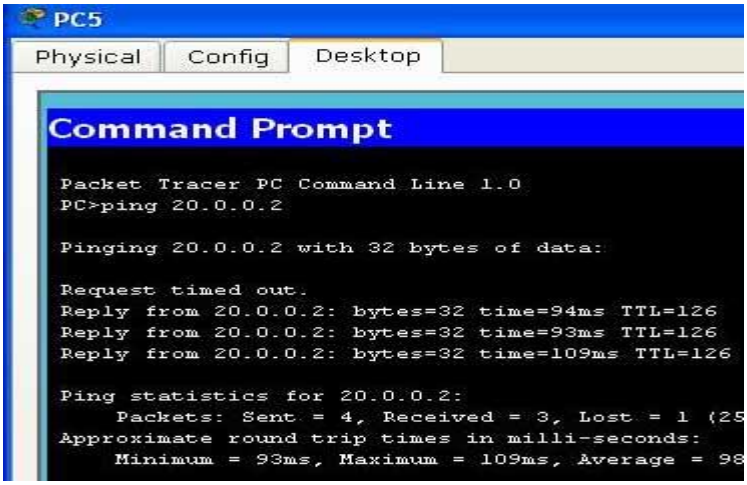
Router(config)#ip route 0.0.0.0 0.0.0.0 serial 0/0/0

Router(config)#hostname R2

ដើម្បីតេស្តទៅលើ NAT នៅលើ R1 ប្រើបញ្ជា

R1#debug ip nat

នៅលើ PC ប្រើ ping មកកាន់ 20.0.0.2



នៅពេលដែល ICMP ping packet មកដល់ R1។ វាពិនិត្យមើលទៅលើ Address របស់ Source របស់វាប្រៀបធៀបជាមួយ access list 1។ នៅពេលដែល Packet នេះត្រូវបានបង្កើតពីណេតវើក192.168.0.0 នោះវានិងឆ្លងកាត់ Access list។ ឥឡូវនេះ Router ពិនិត្យទៅលើ NAT pools ដើម្បីបកប្រែ។ នៅពេលអ្នកពិនិត្យនៅក្នុងលទ្ធផលរបស់ debug នៅលើ R1។

IP NAT debugging is on

NAT: s=192.168.0.7->50.0.0.1, d=20.0.0.2[1]

NAT*: s=20.0.0.2, d=50.0.0.1->192.168.0.7[1]

NAT: s=192.168.0.7->50.0.0.1, d=20.0.0.2[1]

NAT*: s=20.0.0.2, d=50.0.0.1->192.168.0.7[1]

NAT: s=192.168.0.7->50.0.0.1, d=20.0.0.2[1]

NAT*: s=20.0.0.2, d=50.0.0.1->192.168.0.7[1]

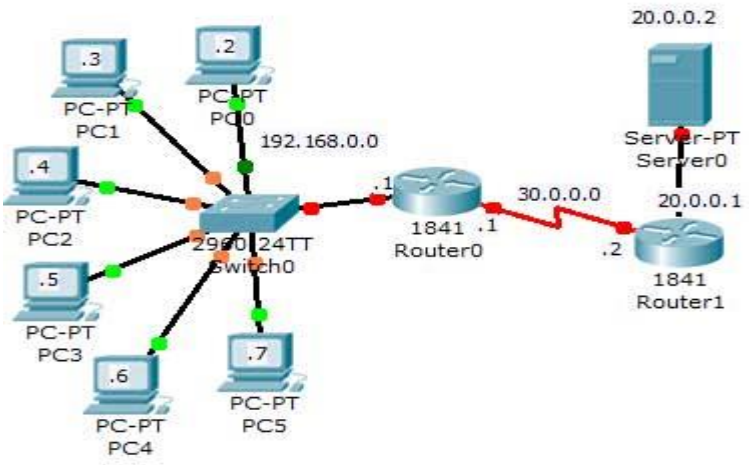
NAT: s=192.168.0.7->50.0.0.1, d=20.0.0.2[1]

NAT*: s=20.0.0.2, d=50.0.0.1->192.168.0.7[1]

ដូចអ្នកបានឃើញលទ្ធផលគឺ 192.168.0.5 ត្រូវបានបកប្រែជាមួយ 50.0.0.1 មុនពេលវាចាកចេញពី Router ។ ឥឡូវពិនិត្យទៅលើ Web access ពី PC។

នៅក្នុងការអនុវត្តជាក់ស្តែងគេត្រូវបិទ debug ក្រោយពីតេស្តរួច។

R1#no debug ip nat IP NAT debugging is off R1#



នៅក្នុង Dynamic NAT ត្រូវបានបង្កើតឡើងរវាង IP និង IP ។ ដូច្នេះអ្នកត្រូវការនូវ Global IP addresses ច្រើននៅពេលដែលអ្នកមាន local address ខាងក្នុង។ បើអ្នកមាន Global IP addresses ពីរឬបីនិងមាន local addresses រាប់រយដែលត្រូវបកប្រែ នោះអ្នកត្រូវប្រើ PAT ។ ចំពោះការបង្ហាញ យើងនឹង Configure ចំពោះ Topology ដែលប្រើ dynamic NAT ប៉ុន្តែនៅពេលនេះយើងប្រើ Global IP address តែមួយគឺ 50.0.0.1 ។

នៅលើ R1

Router>enable

Router#configure terminal Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#hostname R1

R1(config)#interface fastEthernet 0/0

R1(config-if)#ip address 192.168.0.1 255.255.255.0

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#interface serial 0/0/0

R1(config-if)#ip address 30.0.0.1 255.0.0.0

R1(config-if)#clock rate 64000

R1(config-if)#bandwidth 64

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#ip route 0.0.0.0 0.0.0.0 serial 0/0/0

R1(config)#access-list 1 permit 192.168.0.0 0.0.0.255

R1(config)#ip nat pool test 50.0.0.1 50.0.0.1 netmask 255.0.0.0

R1(config)#ip nat inside source list 1 pool test overload

R1(config)#interface fastEthernet 0/0

R1(config-if)#ip nat inside

R1(config-if)#exit

R1(config)#interface serial 0/0/0

R1(config-if)#ip nat outside

R1(config-if)#exit

R1(config)#

នៅលើ R2

Router>enable

Router#configure terminal

Router(config)#interface serial 0/0/0

Router(config-if)#ip address 30.0.0.2 255.0.0.0

Router(config-if)#no shutdown

Router(config-if)#exit

Router(config)#interface fastEthernet 0/0

Router(config-if)#ip address 20.0.0.1 255.0.0.0

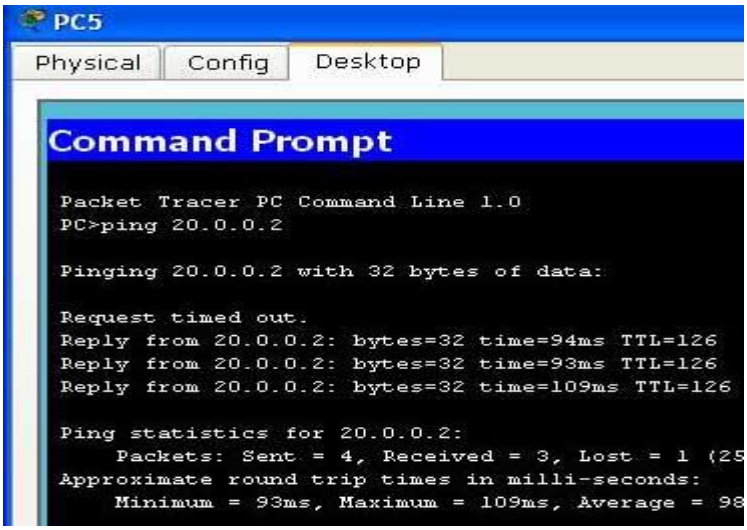
Router(config-if)#no shutdown

Router(config-if)#exit

Router(config)#hostname R2

R2(config)#ip route 0.0.0.0 0.0.0.0 serial 0/0/0

តឡូវនេះ ping មកកាន់ 20.0.0.2



ដើម្បីផ្ទៀងផ្ទាត់ចំពោះ PAT នៅលើ R1 ត្រូវប្រើបញ្ជា show ip nat translations

R1#show ip nat translations

Pro	Inside global	Inside local	Outside local	Outside global
icmp	50.0.0.1:1	192.168.0.7:1	20.0.0.2:1	20.0.0.2:1
icmp	50.0.0.1:2	192.168.0.7:2	20.0.0.2:2	20.0.0.2:2
icmp	50.0.0.1:3	192.168.0.7:3	20.0.0.2:3	20.0.0.2:3
icmp	50.0.0.1:4	192.168.0.7:4	20.0.0.2:4	20.0.0.2:4

ជំពូកទី១២

សេចក្តីផ្តើមចំពោះ Access-Lists នៅលើ Cisco IOS

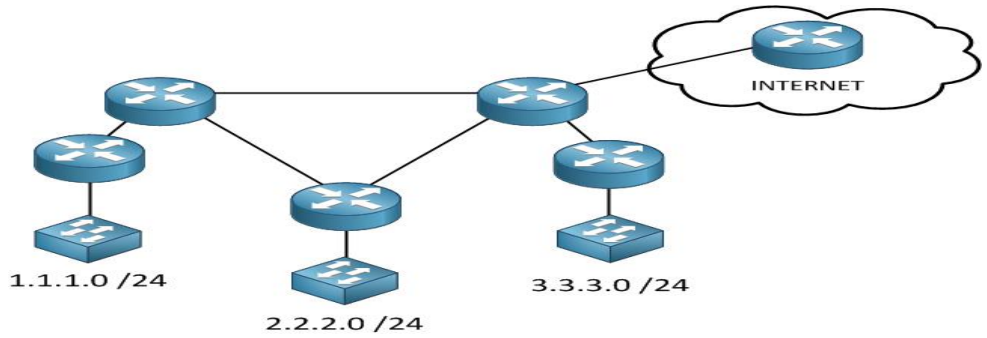
១២-១-Router

នៅក្នុងពិភពលោកមួយដ៏ល្អឥតខ្ចោះគឺជាទីកន្លែងដែលយើងទាំងអស់គ្នាទុកចិត្តគ្នាទៅវិញទៅមកហើយគ្មាននរណាម្នាក់ធ្វើឲ្យមានកំហុសនោះទេ។ ដូច្នេះមិនចាំបាច់ត្រូវការនូវសុវត្ថិភាព។ នៅក្នុងជីវិតពិត បញ្ហាអាក្រក់អាចកើតមានឡើងចំពោះណេតវើករបស់យើងបានគ្រប់ពេល។ ដូច្នេះយើងត្រូវតែការពារវា។ នៅក្នុងជំពូកនេះយើងត្រូវការប្រើនូវ access-lists ហើយយើងនិងសិក្សាពីភាពខុសគ្នារវាង standard និង extended access-lists។

Access-lists ដំណើរការនៅលើស្រទាប់ទី៣ គឺណេតវើកហើយនិង Transport គឺ Layer ទី៤ ហើយត្រូវបានប្រើក្នុងគោលបំណង២ផ្សេងគ្នា:

Filtering

Classification

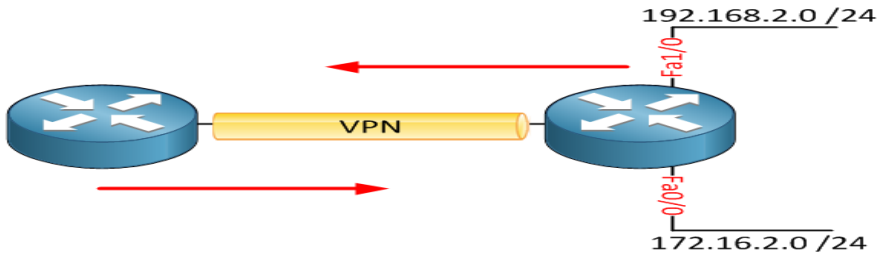


Filtering

ត្រូវបានប្រើដើម្បីអនុញ្ញាតឬបដិសេធចំពោះចរាចរណ៍ដែលអាចមកដល់ផ្នែកណាមួយនៃណេតវើករបស់យើង។ បើគ្មានការច្រោះចំពោះចរាចរណ៍នោះទេ វាអាចទៅកន្លែងណាក៏បាន។ បើអ្នកពិនិត្យមើលទៅលើរូបភាពខាងលើ អ្នកប្រហែលជាមិនចង់ឲ្យ IP packets ពី Internet ចូលមកកាន់ណេតវើករបស់អ្នកនោះទេ។ អ្នកក៏អាចប្រើ Access-list ដើម្បីបិទ IP Packets ពី 3.3.3.0 /24 មកកាន់ 1.1.1.0 /24 ឬ network addressផ្សេងទៀត។

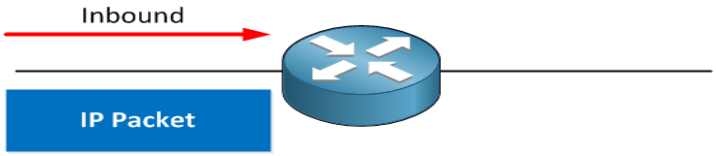
Classification

មិនបោះចោលនូវ IP Packets ដូចជា Filtering នោះទេ ប៉ុន្តែយើងប្រើវាសម្រាប់រកឲ្យឃើញពីចរាចរណ៍។ សូមពិនិត្យមើលទៅលើរូបភាពខាងក្រោម:

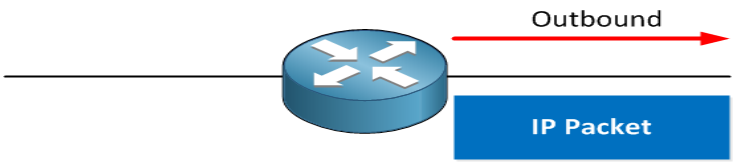


នៅក្នុងឧទាហរណ៍ខាងលើយើងមាន VPN ដែលអាច encrypts ចរាចរណ៍រវាង Router ពីរ។ នៅពេលណាក៏ដោយ យើងអាចបង្កើតបាននូវ VPN មួយដែលយើងអាចប្រើ access-list ដើម្បី"ជ្រើសរើស"អំពីចរាចរណ៍ដែលត្រូវ Encrypt ។ អ្នកប្រហែលជាចង់ឲ្យចរាចរណ៍ពី ណេតវើក192.168.2.0 /24 ដែលត្រូវ Encrypt ។ ប៉ុន្តែចរាចរណ៍ពី 172.16.2.0 /24 មិនត្រូវបាន Encrypt នោះទេ។ យើងអាចប្រើ access-list ដើម្បីជ្រើសរើសចរាចរណ៍ដែលគេហៅថា classification ។

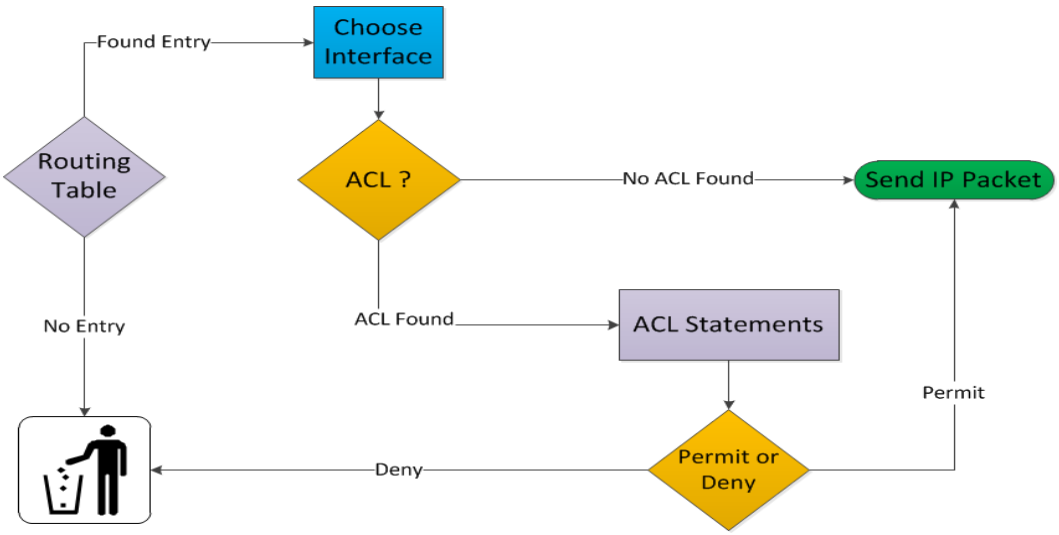
សូមពិនិត្យមើលចំពោះ Filtering ។ ក្រោយពីបង្កើតបាននូវ access-list មួយហើយមាន Spots បីដែលត្រូវដាក់វាគឺ



អ្នកអាចដាក់វាជា inbound នៅលើ Interface ដែលមានន័យថាគ្រប់ Packets ទាំងអស់ដែលមកដល់ Routers របស់អ្នកនិងជួបជាមួយ access-list នឹងត្រូវបានត្រួតពិនិត្យជាមួយ access-list ។



ជម្រើសមួយទៀតគឺដាក់ Access-list ចំពោះ outbound ។ នៅក្នុងករណីនេះ IP Packets នឹងឆ្លងកាត់តាម Router ហើយភ្លាមនោះវានិងចាកចេញពី interface ដែលវាកំពុងត្រូវបានត្រួតពិនិត្យជាមួយ Access-list ។ នៅពេលដែលអ្នកដាក់ access-list ចំពោះ outbound ។ នេះគឺជាអ្វីដែល router របស់អ្នកនិងធ្វើ:



១-IP Packets និងចូលទៅក្នុង router

២-Router និងត្រួតពិនិត្យមើលថាតើវាដឹងអំពីគោលដៅដោយពិនិត្យនៅក្នុង routing table របស់វា

៣-បើគ្មាននៅក្នុង routing table នោះ IP packet និងត្រូវបានបោះចោល

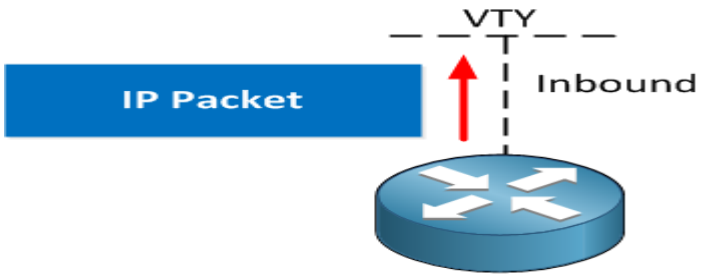
៤-បើមាននៅក្នុង routing table វានិងជ្រើសរើសនូវ Interface ដែលចេញត្រឹមត្រូវ

៥-បើគ្មាន Access-list នោះទេ IP Packet និងត្រូវបានបញ្ជូនចេញក្រៅតាម Interface

៦-បើមាន Access-list យើងនឹងត្រូវពិនិត្យទៅលើ IP Packet ហើយធ្វើការប្រៀបធៀបវាជាមួយ access-list.

៧-បើ IP Packet ត្រូវបានអនុញ្ញាត វានិងត្រូវបានបញ្ជូនបន្ត ។ បើពុំនោះទេ វានិងត្រូវបានបោះចោល

ជម្រើសទី៣គឺប្រើវាជាមួយ VTY line ។ យើងអាចប្រើវាដើម្បីធ្វើឲ្យ Telnetនិង SSH មានសុវត្ថិភាព ។



សូមពន្យល់ពី Access-list ធ្វើការដូចខាងក្រោម:

Router#show access-lists

Standard IP access list 1

10 permit 192.168.1.0, wildcard bits 0.0.0.255

20 permit 192.168.2.0, wildcard bits 0.0.0.255

30 permit 172.16.0.0, wildcard bits 0.0.255.255

Access-lists មិនប្រើ subnet masks នោះទេគឺប្រើ wildcard bits ។ នេះមានន័យថាចំពោះ bit 0 ត្រូវបានជំនួសដោយ 1 និងផ្ទុយមកវិញ ។

សូមបង្ហាញនូវឧទាហរណ៍ដូចខាងក្រោម:

Subnet mask 255.255.255.0 គឺមាន 0.0.0.255 ជា wildcard mask ។ ដើម្បីពន្យល់ចំពោះបញ្ហានេះ យើងត្រូវការបង្ហាញនូវប្រព័ន្ធគោលពីរ ។

Bits	128	64	32	16	8	4	2	1

255	1	1	1	1	1	1	1	1

នេះគឺជា Octet ទី១នៃ subnet mask (255.255.255.0) ជាលេខប្រព័ន្ធគោលពីរ។ ដូចអ្នកបានដឹងហើយថា គ្រប់តម្លៃទាំងអស់មានលេខ១ទាំងអស់ដែលបានជាលេខគោល១០គឺ 255។

Bits	128	64	32	16	8	4	2	1
0	0	0	0	0	0	0	0	0

នេះក៏ជា octet ទី១ផងដែរ។ ប៉ុន្តែមាន wildcard bits។ បើអ្នកចង់បានតម្លៃ Wildcard សមមូល អ្នកគ្រាន់តែត្រឡប់ Bits ប៉ុណ្ណោះ។ មានន័យថាបើមានលេខ១ អ្នកគ្រាន់តែប្តូរវាជា0វិញ។

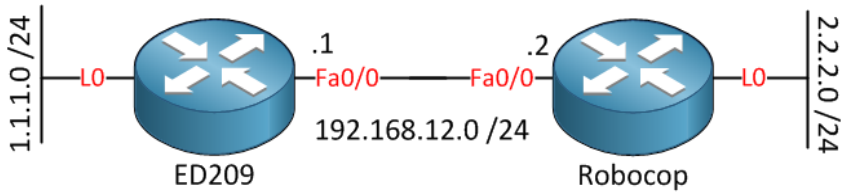
សូមពិនិត្យមើលពី Subnet mask មួយទៀតគឺ 255.255.255.128។ តើវាមាន wildcard សមមូលអ្វី? យើងមាន

Bits	128	64	32	16	8	4	2	1
128	1	0	0	0	0	0	0	0

ដូច្នេះយើងគ្រាន់តែប្តូរ Subnet mask នៃ Octet ចុងក្រោយនោះយើងទទួលបាន។

ឧទាហរណ៍ចំពោះ Standard access-list នៅលើ Cisco Router

សូមពិនិត្យមើលចំពោះ Access-list ខ្លះដែលបានបង្ហាញចំពោះ Cisco IOS routers ។ នៅក្នុងដំពូកនេះយើងនិងសិក្សាទៅលើ Standard access-list ។ នេះគឺជា Topology ដែលយើងសិក្សា។



មាន Routers ពីរហើយ Router នីមួយៗមាន loopback interface។ យើងនិងប្រើ static routesចំនួនពីរ។ ដូច្នេះ Routers អាចស្គាល់គ្នាតាម loopback interface របស់ Router ។

ED209(config)#ip route 2.2.2.0 255.255.255.0 192.168.12.2

Robocop(config)#ip route 1.1.1.0 255.255.255.0 192.168.12.1

បើអ្នកជ្រើសរើសប្រើ routing protocol ដើម្បីផ្សព្វផ្សាយពីណេតវើកត្រូវដឹងថា access-list មិនបិទចំពោះ RIP, EIGRP ឬ OSPF ចារចរណ៍ នោះទេ។

ឥឡូវនេះចាប់ផ្តើមជាមួយ Standard access-list ។ យើងនិងបង្កើតអ្វីមួយនៅលើ router Robocop ដែលអនុញ្ញាតឲ្យចរាចរណ៍ពីណេតវើក192.168.12.0 /24:

Robocop(config)#access-list 1 permit 192.168.12.0 0.0.0.255

មានការអនុញ្ញាតឲ្យតែមួយគត់ ។ ត្រូវចាំថានៅខាងក្រោមនៃ Access-list គឺជា “deny any” ។ យើងមិនបានឃើញវានោះទេ ប៉ុន្តែវានៅទីនោះ ។ ឥឡូវនេះប្រើ access-list inbound ចំពោះ router Robocop:

Robocop(config)#interface fastEthernet 0/0

Robocop(config-if)#ip access-group 1 in

ការប្រើបញ្ជា ip access-group ដើម្បីអនុវត្តជាមួយ Interface មួយ ។ គេបានប្រើពាក្យថា in ។

Robocop#show ip interface fastEthernet 0/0

FastEthernet0/0 is up, line protocol is up

  Internet address is 192.168.12.2/24

  Broadcast address is 255.255.255.255

  Address determined by setup command

  MTU is 1500 bytes

  Helper address is not set

  Directed broadcast forwarding is disabled

  Outgoing access list is not set

  Inbound access list is 1

អ្នកអាចផ្ទៀងផ្ទាត់ថា access-list ត្រូវបានអនុវត្តជាមួយបញ្ជា show ip interface ។ ដូចខាងលើបញ្ជាក់ថា access-list 1 ត្រូវបានប្រើជាមួយ inbound ។

ឥឡូវនេះបង្កើតចរាចរណ៍ខ្លះ

ED209#ping 192.168.12.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:

!!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms

Our ping is successful; let's check the access-list:

Robocop#show access-lists

Standard IP access list 1

10 permit 192.168.12.0, wildcard bits 0.0.0.255 (27 matches)

ដូចអ្នកបានឃើញក្នុងការបង្ហាញ access-list បានឲ្យដឹងពីចំនួននៃអ្វីដែលត្រូវគ្នា។ យើងអាចប្រើវាដើម្បីផ្ទៀងផ្ទាត់ចំពោះ access-list ។សូមបង្ហាញពីអ្វីដែលមានសារៈសំខាន់នៅពេលប្រើ access-lists:

ED209#ping 192.168.12.2 source loopback 0

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:

Packet sent with a source address of 1.1.1.1

U.U.U

Success rate is 0 percent (0/5)

នៅពេលអ្នកប្រើបញ្ជា Ping អ្នកអាចប្រើ keyword ដើម្បីជ្រើសរើសយក Interface ។ source IP address របស់ IP packet នេះគឺ 1.1.1.1 ហើយអ្នកអាចដឹងពី Ping ដែលបរាជ័យដោយសារតែ access-list បានបោះវាចោល។

Robocop#show access-lists

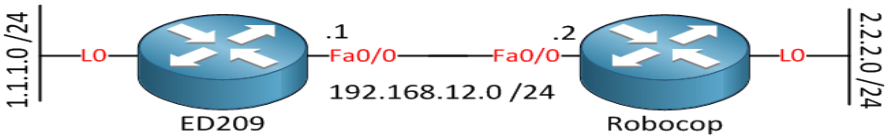
Standard IP access list 1

10 permit 192.168.12.0, wildcard bits 0.0.0.255 (27 matches)

អ្នកមិនបានឃើញវាជាមួយបញ្ជា show access-list ពីព្រោះថា "deny any" កំពុងបោះវាចោល។ តើមានអ្វីកើតឡើងបើអ្នកចង់បានអ្វីដែលប្លែក? ឧបមាថាយើងចង់បដិសេធចោលចំពោះចរាចរណ៍ពីណេតវើក192.168.12.0 /24 ប៉ុន្តែអនុញ្ញាតឲ្យណេតវើកផ្សេងទៀត?

ឧទាហរណ៍ពី Extended Access-List ចំពោះ Cisco Router

យើងមាន Topology ដូចខាងក្រោម:



ការប្រើ extended access-list យើងអាចបង្កើតបាននូវឃ្លាដ៏សំញាំ។ ឧបមាថាយើងមានតម្រូវការដូចខាងក្រោម:

ចរាចរណ៍ពីណេតវើក 1.1.1.0/24 ត្រូវបានអនុញ្ញាតភ្ជាប់មកកាន់ HTTP server នៅលើ router Robocop ។ ប៉ុន្តែវាត្រូវបានអនុញ្ញាតឲ្យភ្ជាប់មកតែ IP address 2.2.2.2 ចំពោះចរាចរណ៍ផ្សេងទៀតត្រូវបានបដិសេធ ។ ឥឡូវនេះអ្នកត្រូវការបកប្រែវាជាមួយ extended access-list ។ វាមាន Syntax ដូចជា:

[source] + [source port] to [destination] + [destination port]

សូមពិនិត្យមើលទៅលើ Configuration:

```
Robocop( config )#access-list 100 ?
  deny     Specify packets to reject
  dynamic  Specify a DYNAMIC list of PERMITs or DENYs
  permit   Specify packets to forward
  remark   Access list entry comment
```

ជាដំបូងយើងត្រូវជ្រើសរើស Permit ឬdeny ។ អ្នកអាចប្រើវាដើម្បីបន្ថែមការបរិយាយចំពោះ access-list ។

```
Robocop( config )#access-list 100 permit ?
  <0-255>  An IP protocol number
  ahp      Authentication Header Protocol
  eigrp    Cisco's EIGRP routing protocol
  esp      Encapsulation Security Payload
  gre      Cisco's GRE tunneling
  icmp     Internet Control Message Protocol
  igmp     Internet Gateway Message Protocol
  ip       Any Internet Protocol
  ipinip   IP in IP tunneling
  nos      KA9Q NOS compatible IP over IP tunneling
  ospf     OSPF routing protocol
  pcp      Payload Compression Protocol
```

pim Protocol Independent Multicast

tcp Transmission Control Protocol

udp User Datagram Protocol

អ្នកបានឃើញថាមាន Options ជាច្រើន។ ដោយសារតែអ្នកចង់អនុញ្ញាតឲ្យតែចរាចរណ៍របស់ HTTP នោះយើងត្រូវតែជ្រើសរើសយក TCP ។

Robocop(config)#access-list 100 permit tcp ?

A.B.C.D Source address

any Any source host

host A single source host

ឥឡូវនេះយើងត្រូវតែជ្រើសរើសយក source។ យើងអាចយកប្រភេទជា network address ដែលមាន wildcard ឬអាចប្រើពាក្យថា anyឬhost keyword។ ពាក្យទាំងពីរនេះគឺជាពាក្យកាត់ សូមពន្យល់ដូចខាងក្រោម:

បើអ្នកវាប្រើ “0.0.0.0 255.255.255.255” អ្នកមានណេតវើកទាំងអស់។ ផ្ទុយទៅវិញអ្នកអាចប្រើពាក្យថា any keyword

បើអ្នកប្រើជា “2.2.2.2 0.0.0.0” នោះវាត្រូវនិង IP address តែមួយ។ ជំនួសឲ្យ “0.0.0.0” wildcard អ្នកអាចប្រើ host

យើងចង់ជ្រើសរើសយកណេតវើក 1.1.1.0 /24 ជា source។ ដូច្នេះអ្វីដែលអ្នកត្រូវធ្វើគឺ:

Robocop(config)#access-list 100 permit tcp 1.1.1.0 0.0.0.255 ?

A.B.C.D Destination address

any Any destination host

eq Match only packets on a given port number

gt Match only packets with a greater port number

host A single destination host

lt Match only packets with a lower port number

neq Match only packets not on a given port number

range Match only packets in the range of port numbers

ក្រៅពីការជ្រើសរើសនូវ source យើងក៏ត្រូវការជ្រើសរើសនូវ source port number។ ត្រូវចាំថានៅពេលភ្ជាប់ពី router ED209 មកកាន់ router Robocop's HTTP server ដែលមានលេខ source port ចៃដន្យ។ នោះយើងមិនកំនត់លេខ Port នៅទីនេះទេ។

Robocop(config)#access-list 100 permit tcp 1.1.1.0 0.0.0.255 host 2.2.2.2 ?

ack Match on the ACK bit

dscp Match packets with given dscp value

eq Match only packets on a given port number

established Match established connections

fin Match on the FIN bit

fragments Check non-initial fragments

gt Match only packets with a greater port number

log Log matches against this entry

log-input Log matches against this entry, including input interface

lt Match only packets with a lower port number

neq Match only packets not on a given port number

precedence Match packets with given precedence value

psh Match on the PSH bit

range Match only packets in the range of port numbers

rst Match on the RST bit

syn Match on the SYN bit

time-range Specify a time-range

tos Match packets with given TOS value

urg Match on the URG bit

<cr>

យើងនិងជ្រើសរើសយកគោលដៅដែលមាន IP address គឺ 2.2.2.2។ យើងប្រើ “2.2.2.2 0.0.0.0” ប៉ុន្តែជាការងាយស្រួលយើងប្រើពាក្យថា host keyword។ ក្រៅពី IP address របស់គោលដៅ យើងអាចជ្រើសរើសយកលេខ Port របស់គោលដៅជាមួយពាក្យថា eq keyword:

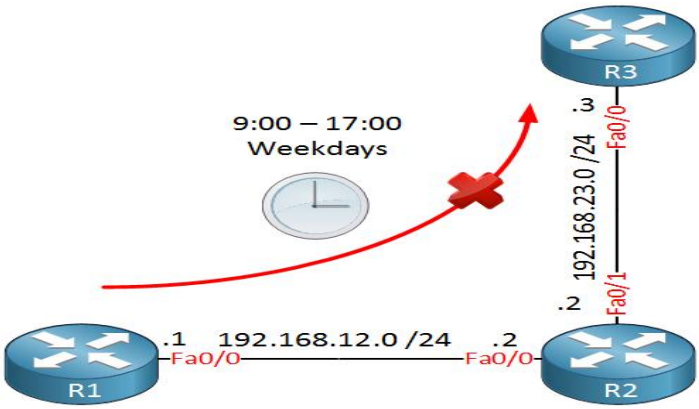
Robocop(config)#access-list 100 permit tcp 1.1.1.0 0.0.0.255 host 2.2.2.2 eq 80

Cisco IOS Time Based Access-List

នៅពេលខ្លះមានសារៈណាស់ដែលត្រូវបិទចំពោះចរាចរណ៍ចំពោះថ្ងៃណាមួយឬម៉ោងពេលធ្វើដំនួញ។ ឧទាហរណ៍ អ្នកចង់បិទចរាចរណ៍ facebook ពីចន្លះដល់សុក្រវាងម៉ោង 9:00–17:00។ យើងអាចទទួលបានដោយប្រើរយៈពេលដោយកំនត់នៅក្នុង access-lists។ នៅពេលអ្នកប្រើវា ឃ្លានៅក្នុង access-list និងមានសកម្មភាពក្នុងកំឡុងពេលនោះដែលអ្នកបានកំនត់។ សូមពិនិត្យមើលឧទាហរណ៍

១២-២-Configuration

ការបង្ហាញពីការប្រើពេលវេលាពឹងផ្អែកទៅលើ access-list ដែលមាន Topology ដូចខាងក្រោម:



ខាងលើមាន Router ចំនួន៣។ ឧបមាថា R1 គឺជា User ម្នាក់នៅលើកុំព្យូទ័រមួយហើយR3គឺជា Web server ។ យើងចង់ការពីការចូលប្រើពី R1 មកកាន់ webserver នៅលើ R3 នៅក្នុងម៉ោងធ្វើការរវាង 9:00 – 17:00។ យើងនឹង Configure ពេលវេលាដោយពឹងផ្អែកទៅលើ access-list នៅលើ R2។ បញ្ជា time range អាស្រ័យទៅលើម៉ោងដែលធ្វើឲ្យពេលវេលានិងកាលបរិច្ឆេទត្រឹមត្រូវ។

R2#clock set 12:48:00 14 July 2015

តាមធម្មតាជាការល្អត្រូវប្រើ NTP ។ ឥឡូវនេះយើងអាច Configure ចំពោះពេលវេលា។

R2(config)#time-range WORK_HOURS

R2(config-time-range)#periodic ?

Friday Friday

Monday Monday

Saturday Saturday

Sunday Sunday

Thursday Thursday

Tuesday Tuesday

Wednesday Wednesday

daily Every day of the week

weekdays Monday thru Friday

weekend Saturday and Sunday

មាន options មួយចំនួនដែលយើងអាចជ្រើសរើសយកថ្ងៃពិតប្រាកដឬអ្នកអាចប្រើដូចជា weekdays,weekend ។

ឧទាហរណ៍

R2(config-time-range)#periodic weekdays 09:00 to 17:00

យើងមានពេលវេលាដែលមានឈ្មោះថា “WORK_HOURS” សម្រាប់ម៉ោងធ្វើការ ។ សូមបង្កើត access-list:

R2(config)#ip access-list extended NO_FACEBOOK

R2(config-ext-nacl)#deny tcp any host 192.168.23.3 eq 80 time-range WORK_HOURS

R2(config-ext-nacl)#permit ip any any

ចំពោះ access-list ខាងលើមានឃ្លាថា blocks ចារចរណ៍ចំពោះ TCP port 80 នៅលើ 192.168.23.3 ។ ប៉ុន្តែមានតែពេលវេលាដែលបានកំនត់ច្បាស់លាស់ ។ សូមធ្វើឲ្យវាមានសកម្មភាពលើ Interface ។

R2(config)#interface FastEthernet 0/0

R2(config-if)#ip access-group NO_FACEBOOK in

ឥឡូវនេះយើងអាចសាកល្បងភ្ជាប់ពី R1 មក R3

R1#telnet 192.168.23.3 80

Trying 192.168.23.3, 80 ...

% Destination unreachable; gateway or host down

យើងមិនអាចភ្ជាប់មកកាន់ Webserver នៅលើ R3 បាននោះទេ ។

ឧទាហរណ៍

បង្កើត Standard IP ACL

ឧបមាថាយើងចង់អនុញ្ញាតឲ្យតែ Host ដែលមាន IP address 20.0.0.10 និង subnetmask: 255.0.0.0 ហើយបិទចោលចំពោះ Hosts ផ្សេងៗទៀត។ ដើម្បីទទួលបានតាមលក្ខខណ្ឌខាងលើ យើងត្រូវបង្កើតនូវលក្ខខណ្ឌពីរសម្រាប់ ACL ។

ទី១: អនុញ្ញាតចំពោះ 20.0.0.10 255.0.0.0.0

ទី២: បិទចំពោះ Hosts ផ្សេងៗទៀត(All)

Router(config)#access-list 10 permit 20.0.0.10 0.0.0.0

Router(config)#access-list 10 deny any

ក្នុងគោលបំណងដើម្បីច្រោះបានជាដំបូងយើងបង្កើតលក្ខខណ្ឌបដិសេធហើយបន្ទាប់មកបិទចោលចំពោះចរាចរណ៍ពីគ្រប់ Hosts ទាំងអស្សរួមមាន 20.0.0.10 ។ ឧទាហរណ៍យើងមានលក្ខខណ្ឌខាងក្រោម:

Router(config)#access-list 10 deny any

Router(config)#access-list 10 permit 20.0.0.10 0.0.0.0

ACL នេះនិងបិទគ្រប់ចរាចរណ៍ពីគ្រប់ Hosts ទាំងអស់។ ហេតអ្វី?

ពីព្រោះថាលក្ខខណ្ឌត្រូវគ្នារាប់ពីលើចុះក្រោមតាមលំដាប់ហើយនៅពេលជួបលក្ខខណ្ឌមួយហើយ នោះគ្មានលក្ខខណ្ឌណាត្រូវបានគិតទៅទៀតនោះទេ។ ចំពោះលក្ខខណ្ឌទី១នៅក្នុង ACL និងត្រូវជាមួយគ្រប់ Packets ទាំងអស់ពីគ្រប់ Hosts រួមទាំង 20.0.0.10។ លក្ខខណ្ឌទី១ជាសកម្មភាពបិទចោល។ នៅក្នុងសកម្មភាពបិទចោលគឺគ្រប់ Packets ត្រូវបានបោះចោលភ្លាម។ ដូច្នោះគ្រប់ Packets ពីគ្រប់ Hosts ទាំងអស់នឹងត្រូវបានបោះចោលជាមួយលក្ខខណ្ឌទី ១។ គ្មាន Packets ណានៅសល់ដើម្បីជួបជាមួយលក្ខខណ្ឌទី២នោះទេ។ ដូច្នេះដើម្បីអនុវត្តចំពោះការច្រោះបានជោគជ័យ យើងត្រូវយល់ពីដំណើរការរបស់វា។

ដើម្បីយល់ពីលំដាប់នៃលក្ខខណ្ឌ យើងត្រូវបង្កើតលក្ខខណ្ឌពីរ បើពុំដូច្នេះទេយើងមិនត្រូវការបង្កើតនូវលក្ខខណ្ឌបិទចោលនោះទេសម្រាប់គ្រប់ចរាចរណ៍។ គេហៅថា Implicit deny ។ ចំពោះតម្រូវការនេះយើងត្រូវការបង្កើតតែលក្ខខណ្ឌមួយប៉ុណ្ណោះ។

Router(config)#access-list 10 permit 20.0.0.10 0.0.0.0

ឬ

Router(config)#access-list 10 permit host 20.0.0.10

ចំពោះ Host តែមួយគត់ យើងប្រើ wildcard 0.0.0.0 ឬ Host

ឥឡូវនេះយើងបានបង្កើត Standard ACL ជាមួយរបៀបចាស់។ នៅដំហានបន្ទាប់យើងអាចបង្កើតវាជាមួយលក្ខខណ្ឌបែបទំនើប។

Router(config)#ip access-list standard Secure_telnet

Router(config-std-nacl)#permit 20.0.0.10 0.0.0.0

Router(config-std-nacl)#exit

Router(config)#

ឬ

Router(config)#ip access-list standard 10

Router(config-std-nacl)#permit 20.0.0.10 0.0.0.0

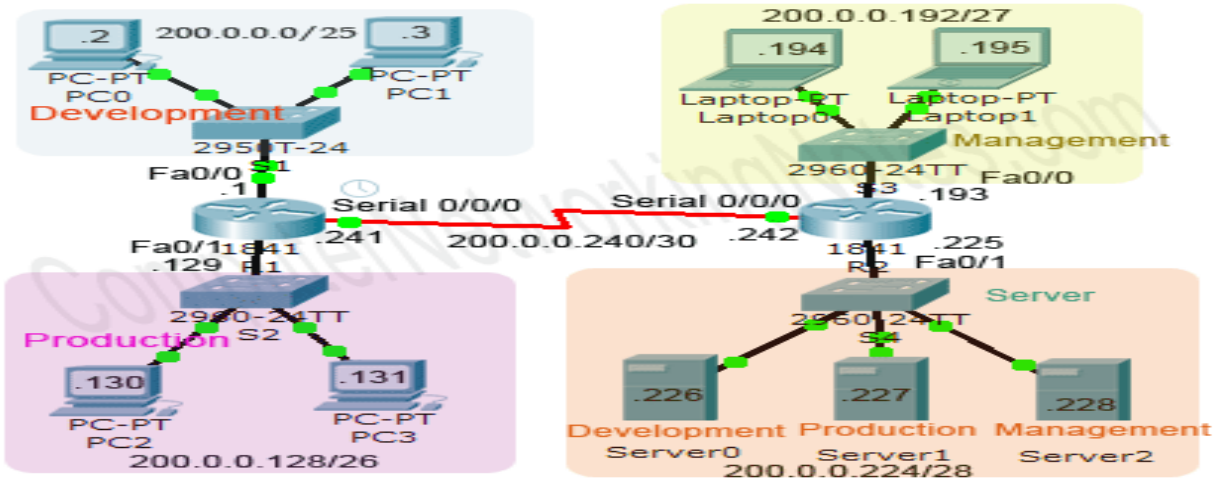
Router(config-std-nacl)#exit

Router(config)#

នៅក្នុងរបៀបទំនើប ការ Configure ខុសពីរបៀបចាស់។ នៅក្នុងរបៀបទំនើប យើងប្រើបញ្ជា ip access-list ជំនួសឲ្យ access-list ។ ប្រាប់ Router ឲ្យដឹងថាយើងកំពុងប្រើ ACL ទំនើប។

ឧទាហរណ៍

នៅក្នុងឧទាហរណ៍នេះបង្ហាញពីការប្រើ Extended Access Control List ។ គេមាន ណេតវើក



Topology diagram ដូចខាងក្រោម:

ចំពោះព័ត៌មានក្បោះក្បាយទាក់ទងជាមួយ Topology នេះមានដូចខាងក្រោម។ ណេតវើក topology នេះត្រូវបាន Configure ជាមួយ

IP address ត្រូវបានកំនត់ទៅឲ្យឧបករណ៍នីមួយៗ

IP addresses ត្រូវបានConfigure នៅលើ Interfaces ដែលបានប្រើ

គេប្រើ RIPv2

នៅក្នុងណេតវើកនេះគេបានភ្ជាប់វាទៅវិញទៅមក។ អ្នកប្រើប្រាស់អាចចូលប្រើបានចំពោះគ្រប់ធនធានពីផ្នែកផ្សេងៗបាន។គេបានជួលអ្នកធ្វើឲ្យ ណេតវើកនេះមានសុវត្ថិភាព។ចំពោះណេតវើកនេះត្រូវការនូវសុវត្ថិភាពដូចខាងក្រោម:

ក្រុមហ៊ុនមាន Servers ចំនួនបី។ បានប្រើ Server មួយសម្រាប់ផ្នែកនីមួយៗគឺ Server0 សម្រាប់ផ្នែកអភិវឌ្ឍន៍ Server1 សម្រាប់ផ្នែកផលិតកម្មនិង Server2 សម្រាប់ផ្នែកគ្រប់គ្រង។ ផ្នែកនីមួយៗអាចប្រើបានតែ Server របស់ខ្លួនឯងតែប៉ុណ្ណោះ។ គេមិនត្រូវបានអនុញ្ញាតឲ្យ Access ចំពោះ Server ផ្សេងបាននោះទេ។ ផ្នែកអភិវឌ្ឍន៍អាចចូលប្រើចំពោះផ្នែកផលិតកម្មបាន។វាមិនអាចចូលប្រើចំពោះផ្នែកគ្រប់គ្រងបាននោះទេ។ ផ្នែកផលិតកម្មអាចចូលប្រើផ្នែកអភិវឌ្ឍន៍បាន។វាមិនអាចចូលប្រើផ្នែកគ្រប់គ្រងបាននោះទេ។

អ្នកប្រើប្រាស់ពីផ្នែកអភិវឌ្ឍន៍មិនត្រូវបានអនុញ្ញាតឲ្យ Ping មកកាន់ Server0 បាននោះទេ។ ប៉ុន្តែវាត្រូវបានអនុញ្ញាតឲ្យចូលប្រើចំពោះ Services ដែលកំពុងដំណើរការនៅលើ Server ។

អ្នកប្រើប្រាស់ពី PC0 ខាងផ្នែកអភិវឌ្ឍន៍មិនត្រូវបានអនុញ្ញាតឲ្យប្រើអ្វីបាននោះទេលើកលែងតែខ្លួនវាផ្ទាល់។ អ្នកប្រើប្រាស់ពី PC2 ត្រូវបានអនុញ្ញាតឲ្យចូលប្រើបានតែ Web server ពី Server។ អ្នកប្រើប្រាស់ពី PC3 ពីផ្នែកផលិតកម្មក៏អាចចូលប្រើចំពោះផ្នែកគ្រប់គ្រងបានដែរ។ អ្នកប្រើប្រាស់ពី laptop0 ពីផ្នែកគ្រប់គ្រងអាចប្រើបានតែServerប៉ុណ្ណោះមិនអាចចូលប្រើផ្នែកអភិវឌ្ឍន៍និងផ្នែកផលិតកម្មបាននោះទេ។ គាត់ត្រូវបានអនុញ្ញាតឲ្យប្រើតែ FTP និង Web service ពី Server ។

ឯកសារយោង

១-Cisco CCNA R&S

២- http://www.cisco.com/en/US/docs/security/asa/roadmap/asaroadmap.html.

៣- http://www.routeralley.com

៤-CCNA Discovery and Explorer (1/2/3/4)

៥-https://networklessons.com