บทที่ 2
เอกสารและงานวิจัยที่เกี่ยวข้อง

ข้อมูลจราจรทางคอมพิวเตอร์

ในการจัดเก็บข้อมูลจราจรทางคอมพิวเตอร์ เมื่อมีการรับส่งข้อมูลกันในระบบเครือข่ายอินเตอร์เน็ตตัวข้อมูล เช่น ข้อความในอีเมล์หรือไฟล์จะถูกทำให้มีขนาดเล็กลงโดยแบ่งออกเป็นส่วนย่อยๆ เรียกว่า Data Packet ซึ่งอยู่ในชั้นเน็ตเวิร์คเลเยอร์ของโอเอสไอโมเดลเป็นเลเยอร์ที่ทำหน้าที่หลักเกี่ยวข้องกับการหาเส้นทาง (Routing) ในการส่งแพ็คเก็ตจากต้นทางไปยังปลายทางโดยส่วนของ Data Packet จะประกอบไปด้วยส่วนของ Header และส่วนของตัวข้อมูล Body โดยในส่วนของ Header จะมีข้อมูลต่างๆ ระบุที่อยู่ปลายทางที่ต้องส่งข้อมูลไป, เลขหมายต้นทางที่ส่งข้อมูลมา, ค่าที่ใช้บอกขนาดความยาว Data Packet, IPv4 และข้อมูลอื่นๆ ซึ่งการในจัดเก็บข้อมูลจราจรทางคอมพิวเตอร์จะใช้หมายเลขไอพีต้นทาง, หมายเลขไอพีปลายทาง, หมายเลขพอร์ต, ชนิดของโปรโตคอล และข้อมูลอื่นๆ

โปรแกรมดักจับแพ็คเก็ตที่เรียกว่า Tcpdump เป็นโปรแกรมประเภทเดียวกับ Sniffer, Wireshark โดยใช้การทำงานแบบ Command-line บนระบบปฏิบัติการลินุกซ์การใช้งานโปรแกรม Tcpdump จะทำการดักจับ (Capture) Traffic หรือ Packet ที่ รับ-ส่ง เข้า-ออก ระหว่างพอร์ตแลน (LAN) ของเครื่องแม่ข่ายที่รันคำสั่งและอุปกรณ์เครือข่าย (Router, Switch, HUB) ซึ่งการเชื่อมต่อเพื่อพิสูจน์ตัวจริงระยะไกลในบริการของผู้ใช้ (Remote Authentication Dial In User Service : RADIUS) เป็นโปรโตคอลแบบไคลเอ็นเซิร์ฟเวอร์ที่ทำงานในชั้นแอพพลิเคชั่นเพื่อรวบรวมแอคเคาท์ของผู้ใช้งานให้อยู่เพียงที่เดียวเพื่อง่ายต่อการบริการไม่ต้องเก็บแอคเคาท์ของผู้ใช้งานหลายจุดหลายเซิร์ฟเวอร์ ถ้ามีผู้ใช้งานที่เซิร์ฟเวอร์ไหนต้องการใช้งานก็จะส่งข้อมูลมาตรวจเช็คที่ RADIUS Server และใช้งานผ่านโปรแกรม Freeradius เป็นตัวจัดการในเรื่องของการพิสูจน์ตัวตนโดยโปรแกรม Freeradius มีหน้าที่หลัก 3 อย่างคือ การตรวจสอบ Username และ Password (Authentication) การกำหนดสิทธิ์ในการเข้าใช้งานของผู้ใช้งาน (Authorization) การเก็บข้อมูลรายละเอียดการใช้งานของผู้ใช้ (Accounting) ซึ่งเป็นโปรโตคอลเครือข่ายที่ให้การตรวจสอบ, อนุมัติ และการจัดการการบัญชี (AAA) จากส่วนกลางสำหรับคอมพิวเตอร์ที่เชื่อมต่อและใช้บริการเครือข่าย การจัดเก็บข้อมูลจราจรทางคอมพิวเตอร์ในประเทศต่างๆ ตามสหภาพยุโรปคุ้มครองข้อมูลส่วนบุคคลทิศทางทั้งหลักการเพื่อจำกัด General of Directive 95/46/EC และบทบัญญัติเฉพาะเจาะจงมากขึ้นของ Directive 97/66/EC ส่วนประเทศไทย ตามพระราชบัญญัติว่าด้วยการ

กระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 กำหนดให้เก็บข้อมูลไว้เป็นเวลาไม่น้อยกว่า 90 วัน นับตั้งแต่ข้อมูลเข้าสู่ระบบและสิ้นสุดการใช้บริการ

ในส่วนของทฤษฎีที่เกี่ยวข้องกับวิจัยหลักๆ ผู้ศึกษาจะอธิบายถึงโปรแกรมต่างๆ ที่ใช้ในการพัฒนาโครงงานนี้ รวมไปถึงพระราชบัญญัติ ว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 เรื่องเกี่ยวกับการเก็บบันทึกข้อมูลจราจรหรือข้อมูลล็อก ซึ่งมีรายละเอียดดังต่อไปนี้

พระราชบัญญัติ ว่าด้วยการกระทำ ความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ.2550

พระราชบัญญัติ หรือ พ.ร.บ. คือ บทกฎหมายที่ใช้บังคับอยู่เป็นประจำตามปกติ เพื่อวางระเบียบบังคับความประพฤติของบุคคล รวมทั้ง องค์กรและเจ้าหน้าที่ของรัฐ ก่อนประกาศใช้บังคับ

การตราพระราชบัญญัตินั้นจะทำได้ก็แต่โดยคำแนะนำและยินยอมของรัฐสภา และเมื่อพระมหากษัตริย์ได้ทรงลงพระปรมาภิไธย และประกาศในพระราชกิจจานุเบกษาแล้ว จึงจะมีผลใช้บังคับเป็นกฎหมายได้

ที่มาของ พ.ร.บ.ว่าด้วยกำรกระทำ ความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 พระราชบัญญัติว่าด้วยการกระทำความผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 เป็นพระราชบัญญัติที่ประกาศลงในราชกิจจานุเบกษา เมื่อวันที่ 18 มิถุนายน พ.ศ. 2550 และมีผลบังคับใช้ตั้งแต่วันที่ 18 กรกฎาคม พ.ศ. 2550 เหตุผลในการประกาศพระราชบัญญัติฉบับนี้ตามข้อความในพระราชกิจจานุเบกษา ระบุว่า เนื่องจากในปัจจุบันระบบคอมพิวเตอร์ได้เป็นส่วนสำคัญของการประกอบกิจการและการดารงชีวิตของมนุษย์ หากมีผู้กระทำด้วยประการใดๆ ให้ระบบคอมพิวเตอร์ไม่สามารถทางานตามคำสั่ง ที่กำหนดไว้ หรือทำให้การทำงานผิดพลาดไปจากคำสั่ง ที่กำหนดไว้ หรือ ใช้วิธีการใด ๆ เข้าล่วงรู้ข้อมูล แก้ไข หรือทำลาย ข้อมูลของบุคคลอื่นในระบบคอมพิวเตอร์โดยมิชอบ หรือใช้ระบบคอมพิวเตอร์เพื่อเผยแพร่ข้อมูลคอมพิวเตอร์อันเป็นเท็จหรือมีลักษณะลามกอนาจาร ย่อมก่อให้เกิดความเสียหาย กระทบกระเทือนต่อเศรษฐกิจสังคม และ ความมั่นคงของรัฐ รวมทั้งความสงบสุขและศีลธรรมอันดีของประชาชน สมควรกำหนดมาตรการเพื่อป้องกันและปราบปรามการกระทำดังกล่าว จึงจำเป็นต้องตราพระราชบัญญัตินี้

การประกาศใช้ พระราชบัญญัติว่าด้วยการกระทำความผิดทางคอมพิวเตอร์ พ.ศ. 2550 มีจุดมุ่งหมายทางเทคนิคในการเก็บข้อมูลจราจรคอมพิวเตอร์ ทั้งนี้เพื่อให้ผู้ให้บริการในแต่ละประเภทได้เก็บข้อมูลดังกล่าวและสามารถนามาใช้ต่อไปได้ คาว่า “ข้อมูลจราจรคอมพิวเตอร์” และ “ข้อมูลผู้ใช้บริการ” เป็นข้อมูลการบันทึกเหตุการณ์ที่เกิดขึ้นกับระบบคอมพิวเตอร์ ดังนั้นข้อมูลจราจรทางคอมพิวเตอร์นับเป็นพยาน หลักฐานสำคัญในการดาเนินคดี อันเป็นประโยชน์อย่างยิ่ง ต่อการ สืบสวน สอบสวน เพื่อนาตัวผู้กระทำความผิดมาลงโทษ ใน พ.ร.บ.ว่าด้วยการกระทำความผิด

เกี่ยวกับคอมพิวเตอร์ จึงสมควรกำหนดให้ผู้ให้บริการมีหน้าที่เก็บรักษาข้อมูลจราจรทางคอมพิวเตอร์ดังกล่าว

โดยศัพท์ทางเทคนิคเรียกข้อมูลจราจรคอมพิวเตอร์ว่า ข้อมูลล็อก หรือ Log ตามค่านิยามใน “มาตรฐานการรักษาความมั่นคงปลอดภัย ในการประกอบธุรกรรมทางอิเล็กทรอนิกส์ประจาปี 2550” ดังนั้นคาว่า “ข้อมูลล็อก” มีความหมายอันเดียวกับคาว่า “ข้อมูลจราจรคอมพิวเตอร์” และ “ข้อมูลผู้ใช้บริการ”

ตามมาตรา 26 ใน พ.ร.บ.ว่าด้วยการกระทำผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 ซึ่งกล่าวไว้ว่า “ผู้ให้บริการต้องเก็บรักษาข้อมูลจราจรคอมพิวเตอร์ไว้ไม่น้อยกว่าเก้าสิบวัน นับแต่วันที่ข้อมูลนั้นเข้าสู่ระบบคอมพิวเตอร์ แต่ในกรณีจา เป็นพนักงานเจ้าหน้าที่จะสั่ง ให้ผู้ให้บริการผู้ใดเก็บรักษาข้อมูลจราจรคอมพิวเตอร์ไว้เกินเก้าสิบวันแต่ไม่เกินหนึ่งปีเป็นกรณีพิเศษเฉพาะรายและเฉพาะคราวก็ได้ ผู้ให้บริการจึงจาเป็นต้องเก็บข้อมูลจราจรคอมพิวเตอร์เท่าที่จาเป็นเพื่อให้สามารถระบุตัวผู้ใช้บริการ นับตั้งแต่เริ่ม ใช้เริ่ม ใช้บริการและต้องเก็บรักษาไว้เป็นเวลาไม่น้อยกว่าเก้าสิบวัน ตั้งแต่การให้บริการสิ้นสุดลง ความในวรรคหนึ่งจะใช้กับผู้ให้บริการประเภทใด อย่างไร และเมื่อใด ให้เป็นไปตามที่รัฐมนตรีประกาศในราชกิจจานุเบกษา ผู้ให้บริการผู้ใดไม่ปฏิบัติตามมาตรานี้ ต้องระวางโทษปรับไม่เกินห้าแสนบาท” นั้น

โดยที่องค์กรแต่ละองค์กรมีลักษณะการใช้ระบบสารสนเทศที่แตกต่างกัน ดังนั้นกระทรวงเทคโนโลยีสารสนเทศและการสื่อสาร จึงได้ออกประกาศกระทรวงฯ สาหรับหลักเกณฑ์ในการจัดเก็บข้อมูลจราจรคอมพิวเตอร์ดังกล่าว เพื่อเป็นแนวทางหรือ “Guideline” ในการจัดเก็บข้อมูลจราจรคอมพิวเตอร์ให้ถูกต้องและเหมาะสมกับลักษณะการใช้ระบบสารสนเทศหรือระบบอินเตอร์เน็ตของแต่ละองค์กรที่มีความแตกต่างกันค่อนข้างมาก เพื่อให้เกิดความเหมาะสมในทางปฏิบัติและเพื่อให้หลายองค์กรได้มีแนวทางที่ชัดเจนในการปฏิบัติ ว่าข้อมูลจราจรอะไรบ้างที่ควรจัดเก็บตลอดจนวิธีการจัดเก็บอย่างถูกต้องตามลักษณะการใช้งานระบบสารสนเทศหรือระบบอินเตอร์เน็ตของแต่ละองค์กร เช่น การจัดเก็บในลักษณะ “Centralized Log” เป็นต้น

มาตรา 5 ถึง มาตรา 10 เป็นการนาหลักการ CIA (Confidentially, Integrity, Availability) มาประยุกต์ใช้กับการโจมตีของผู้ไม่หวังดีในแบบต่าง ๆ เช่น การเจาะระบบเข้าไปขโมยสำเนาข้อมูล การแอบดูชื่อและรหัสผ่าน โดยใช้โปรแกรมประเภท Sniffer หรือการโจมตีเปลี่ยนหน้าเว็บไซต์ (Web Defacement) ตลอดจนการโจมตี ให้เว็บไซต์ล่ม (Denial Of Service) ล้วนแต่เจ้าข่ายมาตรา 5 ถึง มาตรา 10 ซึ่งมีโทษทั้ง จำ และปรับ

มาตรา 11 เกี่ยวข้องกับ “Spam Mail” โดยไม่ระบุชื่อผู้ส่ง ซึ่งมีโทษปรับไม่เกินหนึ่งแสนบาท

มาตรา 12 เป็นการกระทำผิดที่มีโทษในกรณีที่มีผลกระทบต่อความมั่นคงทางเศรษฐกิจของประเทศ หรือ บริการสาธารณะ โทษสูงสุดจา คุก 15 ปี และปรับ 3 แสนบาท แต่ถ้าหากทำให้ผู้อื่นถึงแก่ความตาย โทษจะสูงสุดจา คุกถึง 20 ปี

มาตรา 14 ถึง 16 จะกล่าวถึง ผู้ใช้คอมพิวเตอร์ ตลอดจนผู้ให้บริการ ตามข้อกำหนดของกฎหมาย ต้องระมัดระวังไม่ให้ข้อมูลอันไม่เหมาะสม ปรากฏอยู่บนอินเตอร์เน็ตในลักษณะการปรากฏของตัวข้อมูลเอง เช่น รูปภาพ หรือข้อความ ที่ถูกอัพโหลดขึ้นไป รวมถึง Link ที่จะพาไปยังข้อมูลดังกล่าว โดยมาตรานี้จะเกี่ยวข้องกับ Forward mail หรือ Webboard ซึ่งมีโทษจำ คุกไม่เกิน 5 ปี หรือปรับไม่เกิน 1 แสนบาท หรือ ทั้ง จำ ทั้ง ปรับ

ตามที่ได้วิเคราะห์หมวด 1 เพิ่มเติม ได้ประกาศลงในราชกิจจานุเบกษา เรื่องหลักเกณฑ์ทางเทคนิคในการเก็บข้อมูลจราจรทาง “ความผิดเกี่ยวกับคอมพิวเตอร์” ตั้งแต่มาตรา 5 ถึงมาตรา 17 โดยที่มาตรา 5 ถึง มาตรา 10 และ มาตรา 12 กล่าวถึง ความผิดที่กระทำต่อคอมพิวเตอร์ และมาตรา 11 มาตรา 13 ถึง มาตรา 16 กล่าวถึงการใช้คอมพิวเตอร์ในการกระทำความผิด รวมถึงมาตรา 17 ที่กล่าวถึงการกระทำความผิดนอกราชอาณาจักรต้องรับโทษในราชอาณาจักรนั้น สามารถสรุปสาระสำคัญของมาตรา 5 ถึง มาตรา 17 รวม 13 มาตรา ได้ดัง ตารางที่ 2.1 ดังนี้

ตารางที่ 2.1 หมวด 1 ความผิดเกี่ยวกับคอมพิวเตอร์

ฐานความผิดและบทลงโทษสำหรับการกระทำ โดยมิชอบ
มาตรา 5 การเข้าถึงระบบคอมพิวเตอร์
มาตรา 6 การล่วงรู้มาตรการป้องกันการเข้าถึง
มาตรา 7 การเข้าถึงข้อมูลคอมพิวเตอร์
มาตรา 8 การดักข้อมูลคอมพิวเตอร์โดยมิชอบ
มาตรา 9 การรบกวนข้อมูลคอมพิวเตอร์
มาตรา10 การรบกวนระบบคอมพิวเตอร์
มาตรา 11 สแปมเมล์ (Spam mail)
มาตรา 12 การกระทำความผิดต่อความมั่นคง
มาตรา 13 การจำ หน่ายหรือเผยแพร่ ชุดคา สั่ง เพื่อใช้กระทำความผิด
มาตรา 14 การปลอมแปลงข้อมูลคอมพิวเตอร์หรือเผยแพร่เนื้อหาอันไม่เหมาะสม
มาตรา 15 ความรับผิดชอบของผู้ให้บริการ
มาตรา 16 การเผยแพร่ภาพจากการตัดต่อหรือดัดแปลง
มาตรา 17 การกระทำ ความผิดตามพระราชบัญญัตินี้นอกราชอาณาจักร

ตารางที่ 2.2 บทกำหนดโทษ

ฐานความผิดและบทลงโทษสำหรับการกระทำโดยมิชอบ	**โทษจำ คุก**	**โทษปรับ**
มาตรา 5 การเข้าถึงระบบคอมพิวเตอร์	ไม่เกิน 6 เดือน	ไม่เกิน 10,000 บาท
มาตรา 6 การล่วงรู้มาตรการป้องกันการเข้าถึง	ไม่เกิน 1 ปี	ไม่เกิน 20,000 บาท
มาตรา 7 การเข้าถึงข้อมูลคอมพิวเตอร์	ไม่เกิน 2 ปี	ไม่เกิน 40,000 บาท
มาตรา 8 การดักข้อมูลคอมพิวเตอร์โดยมิชอบ	ไม่เกิน 3 ปี	ไม่เกิน 60,000 บาท
มาตรา 9 การรบกวนข้อมูลคอมพิวเตอร์	ไม่เกิน 5 ปี	ไม่เกิน 100,000 บาท
มาตรา 10 การรบกวนระบบคอมพิวเตอร์	ไม่เกิน 5 ปี	ไม่เกิน 100,000 บาท
มาตรา 11 สแปมเมล์		ไม่เกิน 100,000 บาท
มาตรา 12 การกระทำ ความผิดต่อความมั่นคง 12.1 ก่อความเสียหายแก่ข้อมูลคอมพิวเตอร์ 12.2 กระทบต่อความมั่นคงปลอดภัยของประเทศหรือเศรษฐกิจ วรรคท้าย เป็นเหตุให้ผู้อื่นถึงแก่ชีวิต	ไม่เกิน 10 ปี 3 ปี ถึง 15 ปี 10 ปี ถึง 20 ปี	ไม่เกิน 200,000 บาท 60,000 - 300,000 บาท
มาตรา 13 การจำหน่ายหรือเผยแพร่ ชุดคา สั่ง เพื่อใช้กระทำ ความผิด	ไม่เกิน 1 ปี	ไม่เกิน 20,000 บาท
มาตรา 14 การปลอมแปลงข้อมูลคอมพิวเตอร์หรือเผยแพร่เนื้อหาอันไม่เหมาะสม	ไม่เกิน 5 ปี	ไม่เกิน 100,000 บาท
มาตรา 15 ความรับผิดชอบของผู้ให้บริการ	ไม่เกิน 5 ปี	ไม่เกิน 100,000 บาท
มาตรา 16 การเผยแพร่ภาพจากการตัดต่อหรือดัดแปลง	ไม่เกิน 3 ปี	ไม่เกิน 60,000 บาท

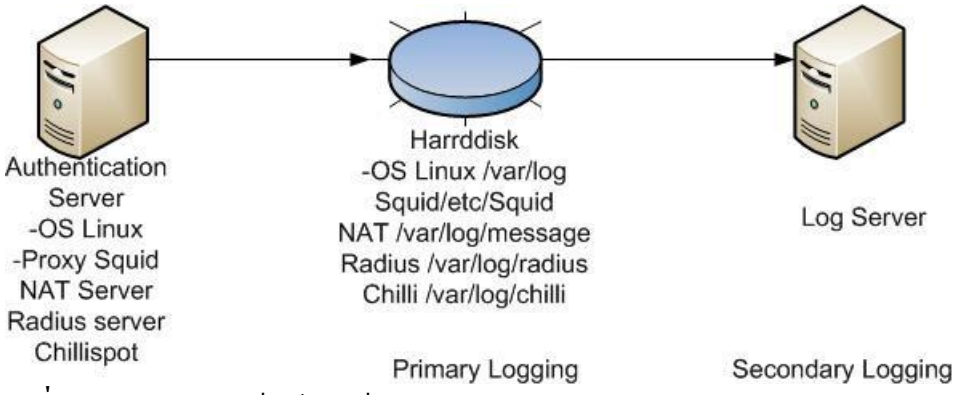
ส่วนประกอบของระบบเก็บข้อมูลจราจรทางคอมพิวเตอร์

1. Log Generation หรือ Log Source เป็นแหล่งกำเนิดข้อมูลล็อกหรือสร้างข้อมูลล็อกเป็นเซิร์ฟเวอร์หรืออุปกรณ์บนเครือข่ายที่มีข้อมูลล็อกจากระบบปฏิบัติการ และแอพพลิเคชั่น การจัดเก็บข้อมูลล็อกบนเครื่องเซิร์ฟเวอร์หรืออุปกรณ์ในตัวเองเรียกว่า Primary Logging ในกรณีที่มีการจัดส่งข้อมูลล็อกไปยังล็อกเซิร์ฟเวอร์ (Log Server) จะเรียกลักษณะการส่งข้อมูลล็อกนี้ว่า Secondary Logging

2. Log Storage and Correlation เป็นล็อกเซิร์ฟเวอร์สำหรับรับข้อมูลล็อกจากแหล่งกำเนิดข้อมูลล็อก (Log Generation) เพื่อจัดเก็บตามรูปแบบที่กำหนดไว้ รวมทั้ง การแปลงข้อมูลล็อกให้อยู่ในรูปแบบที่สามารถจัดเก็บได้ ซึ่งอาจรวมถึงการแปลงรูปแบบข้อมูลล็อกให้พร้อมจะนำไป

วิเคราะห์ต่อได้ ไม่ว่าจะมีรูปแบบของข้อมูลล็อกแตกต่างกัน ในกรณีที่เซิร์ฟเวอร์ดังกล่าวรับข้อมูลล็อกจากแหล่งกำเนิดข้อมูลล็อกจานวนมากจะเรียกว่า Collector หรือ Aggregators

3. Log Analysis and Monitoring เป็นหน้าต่างสาหรับผู้ดูแลระบบ หรือผู้ที่มีหน้าที่รับผิดชอบในการวิเคราะห์ข้อมูลล็อก และติดตามตรวจสอบความถูกต้องของข้อมูลล็อกระบบจัดเก็บข้อมูลล็อกบางระบบสนับสนุนการสร้างรายงานการวิเคราะห์ข้อมูลล็อกทั้งนี้เพื่อให้ข้อมูลเร็วและตรงกับความเป็นจริงในปัจจุบันที่สุด ดังภาพที่ 2.1



ภาพที่ 2.1 แสดงการจัดเก็บข้อมูลล็อกแบบ Primary Logging และ Secondary Logging

ฟังก์ชั่นการทำงานล็อกเซิร์ฟเวอร์ ถ้าพิจารณาตาม พ.ร.บ. ว่าด้วยการกระทำผิดเกี่ยวคอมพิวเตอร์ พ.ศ. 2550 แล้วเจตนารมณ์การจัดเก็บข้อมูลจราจรคอมพิวเตอร์และข้อมูลผู้ใช้ต้องการให้ข้อมูลล็อกนั้นมีความถูกต้องน่าเชื่อถือได้ โดยมีการกำหนดมาตรฐานป้องกันข้อมูลล็อก, มีการกำหนดให้รักษาระยะเวลาเก็บข้อมูลล็อกให้เหมาะสม เช่น 90 วันเป็นอย่างน้อย, สามารถวิเคราะห์ข้อมูลหาผู้ที่เกี่ยวข้องกับระบบสารสนเทศ และรูปแบบของเหตุการณ์ที่เกิดขึ้นจากข้อมูลล็อกได้ นอกจากการเก็บข้อมูลล็อกแล้ว พ.ร.บ.ว่าด้วยการ กระทำความผิดเกี่ยวกับคอมพิวเตอร์ในมาตรา 26 ได้กำหนดให้มีการแต่งตั้ง พนักงานเจ้าหน้าที่ที่มีความเชี่ยวชาญในการวิเคราะห์หลักฐานข้อมูลคอมพิวเตอร์หรือ Computer Forensic ซึ่งต้องดา เนินการเก็บหลักฐานทางอิเล็กทรอนิกส์เพื่อให้สามารถนาไปพิจารณาได้ในชั้นศาลอย่างถูกต้อง หรือ ที่เรียกว่ามีความรู้ความเข้าใจและดาเนินการตามระเบียบวิธีการรักษา Chain of Custody ได้ซึ่งหากพิจารณาแล้วระบบเก็บข้อมูลล็อกตาม พ.ร.บ. จะมีส่วนประกอบดังนี้

ความสามารถทั่วไปของระบบการจัดเก็บข้อมูลล็อก

1. Log Parsing ทำหน้าที่ในการดึงข้อมูลล็อก เพื่อให้สามารถเก็บบนระบบฐานข้อมูลหรือส่งต่อให้กับระบบเก็บฐานข้อมูลล็อกอื่นได้ รวมถึงการแปลงข้อมูลล็อกหรือ LogConversation ให้อยู่ในรูปแบบที่ต้องการหรือพร้อมสาหรับนาไปใช้งานต่อไป

2. Event Filtering ทา หน้าที่กรองข้อมูลล็อก เพื่อใช้สา หรับการวิเคราะห์ การรายงานหรือการประเมินแนวโน้มของเหตุการณ์ ตามคุณลักษณะของเหตุการณ์หรือความผิดปกติที่เกิดขึ้น รวมถึงคัดกรองเหตุการณ์หรือข้อมูลล็อกที่ไม่เกี่ยวข้องฟังก์ชั่น ของ Event Filtering ควรมีการป้องกันการเปลี่ยนแปลงของข้อมูลล็อกด้วย

3. Event Aggregation ทำหน้าที่ในการรวบรวมข้อมูล หรือ ปรับเปลี่ยนข้อมูลให้อยู่ที่เดียวกันเพื่อให้สะดวกต่อการร้องขอหรือสร้างรายงาน

การจัดเก็บข้อมูลจราจรทางคอมพิวเตอร์

1. Log Rotation เป็นการจัดเก็บล็อกไฟล์โดยการหมุนเวียนข้อมูลล็อก หมายถึงการบันทึกไฟล์ข้อมูลไว้เป็นชื่ออื่น และสร้างไฟล์ล็อกใหม่เพื่อรองรับการบันทึกข้อมูลต่อไปตัวอย่างเช่นการบันทึกไฟล์ล็อกเป็น /var/log/message เมื่อมีการหมุนเวียน จะบันทึกข้อมูลล็อกเป็น /var/log/message.1 และสร้างไฟล์ล็อกใหม่เป็นชื่อ /var/log/message เป็นต้นเพื่อป้องกันไม่ให้มีไฟล์ข้อมูลล็อกขนาดใหญ่เกินจนไม่สามารถใช้งานได้ โดยปกติการหมุนข้อมูลล็อกจะดา เนินการตามระยะเวลาที่เหมาะสมเช่น ทุกวัน ทุกสัปดาห์ หรือ เมื่อขนาดของไฟล์ข้อมูลล็อกมีขนาดถึงที่กำหนดไว้ นอกจากนี้ยังนาข้อมูลล็อกเดิมเมื่อมีการหมุนเวียนข้อมูลล็อกไปบีบอัดข้อมูลเพื่อเพิ่มพื้นที่เก็บข้อมูล หรือ ทำ Log archive ได้ การหมุนเวียนข้อมูลล็อกที่เหมาะสมคือการบันทึกข้อมูลล็อกแยกเป็นรายวัน และแยกตามเซิร์ฟเวอร์ หรือ อุปกรณ์ในระบบเครือข่าย

2. Log Archival คือการสา รองข้อมูลล็อกเพื่อให้สามารถรักษาระยะเวลาในการจัดเก็บข้อมูลล็อกตามความต้องการ โดยการบันทึกข้อมูลล็อกบนสื่อบันทึกข้อมูลภายนอก หรือการบันทึกข้อมูลบน SAN (Storage Area Network) หรือการบันทึกบนเซิร์ฟเวอร์หรือข้อมูลที่ทำหน้าที่เฉพาะในการบันทึกข้อมูลล็อก เป็นต้น การจัดทา Log Archival แบ่งเป็นสองแบบคือ

- Log Retention เป็นการบันทึกข้อมูลล็อกของเหตุการณ์จากระบบสม่ำเสมอ
- Log Preservation เป็นกระบวนการรักษาข้อมูลล็อก เพื่อให้สามารถนาไปใช้ร่วมกับการรับมือเหตุการณ์ด้านความมั่นคงปลอดภัยหรือเหตุการณ์ผิดปกติที่เกิดขึ้นกับระบบสารสนเทศและสามารถรักษาข้อมูลล็อกได้ตามระยะเวลากำหนดไว้ หรือตามความต้องการจากภายนอกเช่น ความต้องการ ของ พ.ร.บ. ว่าด้วยการะกระทำผิดเกี่ยวกับคอมพิวเตอร์เป็นต้น

3. Log Compression คือการบีบอัดข้อมูลล็อก เพื่อเพิ่ม พื้นที่ในการจัดเก็บข้อมูลล็อกและง่ายต่อการสำรองข้อมูลล็อก หรือ การย้ายข้อมูลล็อกไปเก็บไว้บนสื่อบันทึกข้อมูลอื่น มักดำเนินการต่อเนื่องจาก Log Rotation หรือ Log archival

4. Log Reduction เป็นการตัด ลบ หรือ ลดข้อมูลล็อกบางส่วนที่ไม่เกี่ยวข้อง เช่นการลบตัวอักษรหรืออักขระที่ไม่จา เป็นต่อการเก็บบันทึกข้อมูลล็อก มักจะดา เนินการควบคู่กับกระบวนการ Log Archival เพื่อลดข้อมูลล็อกที่ไม่เกี่ยวข้องก่อนจะบันทึกข้อมูลล็อกในสื่อบันทึกข้อมูล

5. Log Conversion เป็นการแปลงรูปแบบการจัดเก็บข้อมูลล็อก หรือ แปลงรูปแบบการเก็บข้อมูลล็อกจากรูปแบบหนึ่ง ไปเป็นอีกรูปแบบหนึ่ง เช่น แปลงข้อมูลล็อกจากรูปแบบของไฟล์ Text เป็นรูปแบบข้อมูลล็อกแบบ XML เป็นต้น ส่วนหนึ่งแล้วการทา Log conversion มักจะทำกระบวนการ Event Filtering และ Event Aggregation จนถึง Log Normalization

6. Log Normalization เป็นการปรับรูปแบบของข้อมูลล็อกให้อยู่ในรูปแบบเดียวกัน เช่น การปรับรูปแบบของวันที่ ที่แตกต่างกัน หรือ ความแตกต่างของชื่อตาแหน่งของข้อมูลล็อก มีความสำคัญมากกับการใช้ล็อกเซิร์ฟเวอร์แบบศูนย์กลาง เพื่อเก็บข้อมูลล็อก และ สามารถวิเคราะห์ข้อมูลล็อก ซึ่งต้องมีความสามารถในการรับข้อมูลล็อกหลายรูปแบบ และต้องทำ Log Normalization ในการแปลงข้อมูลล็อกให้อยู่ในรูปแบบที่สามารถจัดเก็บ สืบค้น และวิเคราะห์ได้โดยผู้ที่มีความรู้ความเชี่ยวชาญต่อไป

7. Log File Integrity Checking เป็นกระบวนการตรวจสอบความถูกต้องของล็อกไฟล์โดยการทา Data Hashing กับล็อกไฟล์ที่ไม่มีการเขียนข้อมูลแล้ว เช่น การทำ Log rotation เป็นรายวัน ดังนั้นสามารถนาข้อมูลล็อกไฟล์ของเดือนก่อนหน้ามาเข้ากระบวนการนี้ได้ ซึ่งจะนำมาบีบอัดและคำนวณด้วยวิธี Message Digest เช่นการคำนวณด้วยอัลกอริทึม MD5 ขนาด 128 บิต หรืออัลกอริทึม SHA-1 ขนาด 128 บิต เป็นต้น ผลลัพธ์ที่ได้จะมีขนาดความยาวขนาด 128 บิต เพื่อใช้เป็นตัวแทนของล็อกไฟล์ และควรจัดเก็บไว้ในสื่อบันทึกข้อมูลที่ปลอดภัยเช่น สื่อบันทึกที่เขียนได้อย่างเดียว

ซึ่งในงานวิจัยนี้จะมี แหล่งกำเนิดข้อมูลล็อกที่สำคัญต้องเก็บไว้ตามหลัก พ.ร.บ. คือ ข้อมูลล็อกการเข้าใช้งานระบบ หรือ Account Information เป็นการบันทึกข้อมูลล็อกการพิสูจน์ตัวตนทั้งในกรณีสำเร็จและไม่สำเร็จที่เกิดขึ้น, ข้อมูลการเริ่มใช้งานระบบและเลิกใช้งานของระบบ, ข้อมูลจากเซิร์ฟเวอร์สำหรับการพิสูจน์ตัวตน หรือ Authentication Server ด้วย Protocol RADIUS ซึ่งได้แก่ ล็อก, บัญชีผู้ใช้, รหัสผ่าน, สถานะการพิสูจน์ตัวตน, วันและเวลา เป็นต้น ซึ่งในงานวิจัยนี้มีการใช้งาน Authentication Gateway ในระบบเครือข่ายไร้สายด้วย ซึ่งเรียกว่า Captive Portal เพื่อ

เข้าถึงเครือข่ายไร้สายแบบ Hotspot ข้อมูลล็อกโดยมากจะประกอบด้วย วันเวลาของเครื่องผู้ใช้ หมายเลขไอพีแอดเดรสก่อนและหลังการเชื่อมต่อสถานะการพิสูจน์ตัวตน ชื่อผู้ใช้สำหรับพิสูจน์ตัวตน เป็นต้น

ซอฟต์แวร์ที่เกี่ยวข้องจากข้อมูลขั้น ต้นที่ได้กล่าวถึงเครื่องมือและโปรแกรมต่าง ๆ ที่ใช้ซึ่งอาจจะมีโปรแกรมอื่น ๆ ที่ไม่ได้กล่าวถึง โดยในที่นี้ ผู้ศึกษาจะนาเอา Tools ที่ไม่ใช่ Commercial หรือ Open source มาใช้ในการพัฒนาใช้ให้สอดคล้องกับพระราชบัญญัติว่าด้วยการกระทำผิดเกี่ยวกับคอมพิวเตอร์ พ.ศ. 2550 โดย Software ที่ใช้มีดังต่อไปนี้ Centos 5.9, Radius, NTP (Network Time Protocol), Tcpdump, MySQL 5.6

Remote Authentication Dial-In User Service

RADIUS เป็นคำย่อของ Remote Authentication Dial-In User Service (RADIUS) คือ Client/Server Security Protocol ซึ่งเป็นผลงานของ Lucent InterNetworking Systems ที่ได้ทำการคิดค้นขึ้นมา เพื่อรวบรวม Account ของ Users ให้อยู่แต่เพียงที่เดียว เพื่อง่ายต่อการบริหาร ไม่ต้องเก็บ Account ของ User หลายจุดหลายเซิร์ฟเวอร์ เวลามี users ที่เซิร์ฟเวอร์ไหนต้องการใช้งาน ก็จะส่งข้อมูลมาตรวจเช็คที่ RADIUS server ที่ทำขึ้น ใช้โปรแกรม Freeradius เป็นตัวจัดการในเรื่องของการพิสูจน์ตัวตน โดยโปรแกรม Freeradius มีหน้าที่หลัก 3 อย่างคือ การตรวจสอบ Username และ Password (Authentication) การกำหนดสิทธิ์ ในการเข้าใช้งานของผู้ใช้ (Authorization) การเก็บข้อมูลรายละเอียดการใช้งานของผู้ใช้ (Accounting)

RADIUS Server มีคุณสมบัติ เป็นข้อมูลที่ส่งหรือรับกันระหว่าง RADIUS Server และ RADIUS Client อยู่ในรูปแบบของการร้องขอและตอบกลับ (Request /Response) คือ RADIUS Client ส่งการร้องขอไปยัง RADIUS Server และ RADIUS Server ตอบกลับการร้องขอของ RADIUS Client แต่ละ Package จะต้องระบุจุดประสงค์ของการติดต่อ คือ Authentication หรือ Accounting แต่ละ Package จะบรรจุข้อมูลที่เรียกว่า Attributes ซึ่งใช้ในการตรวจสอบสิทธิ์กำหนดสิทธิ์ และเก็บบันทึกการใช้งาน โดยมีองค์ประกอบพื้นฐานของ Radius Server

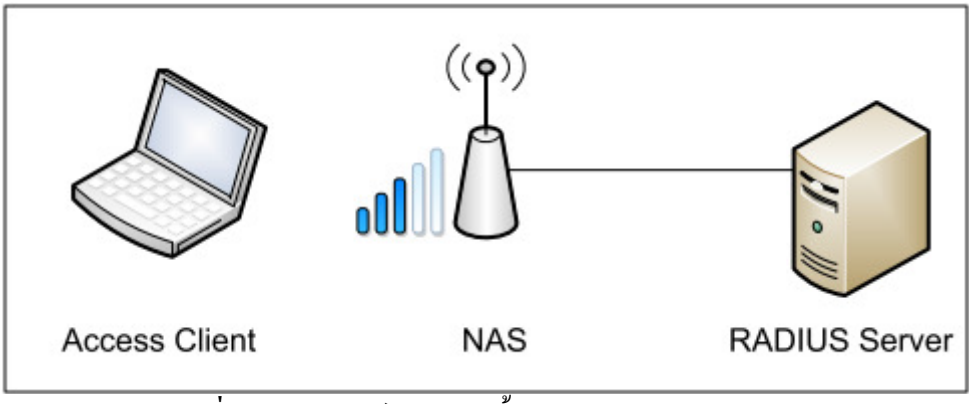
1. Access Clients คือ เครื่องคอมพิวเตอร์หรืออุปกรณ์ที่ผู้ใช้งานสั่งให้ติดต่อระบบเพื่อใช้งาน เช่น เครื่องคอมพิวเตอร์ที่ลูกค้า Individual ใช้งาน โดยใช้ โปรแกรม Dial-Up Net working สั่งงาน Modem ให้ Connect เพื่อใช้งานอินเทอร์เน็ต

2. Network Access Servers (NAS) คือ อุปกรณ์ที่ทำหน้าที่เชื่อมต่อและจัดการการติดต่อระหว่าง Access Clients และ RADIUS Server ซึ่ง NAS จะทำหน้าที่เป็น Client เชื่อมต่อกับ RADIUS Server ส่งผ่านและจัดการข้อมูลที่ใช้ในการตรวจสอบสิทธิ์ กำหนดสิทธิ์ ของ Access

Clients เมื่อ Access Clients ร้องขอการต่อเชื่อมซึ่งจะต้องต่อเชื่อมมายัง NAS ผ่านโพรโตคอลที่ใช้ในการต่อเชื่อมต่าง ๆ เช่น PPP (Point-to-Point Protocol), SLIP (Serial Line Internet Protocol), Extensible Protocol อื่น ๆ เป็นต้น ซึ่งจำเป็นต้องมีการส่งผ่าน Username และ Password จาก Access Clients มายัง NAS หลังจากนั้น NAS จะส่งข้อมูลที่จำเป็นต่าง ๆ เช่น Username, Password, NAS IP Address, NAS Port Number และข้อมูลอื่น ๆ ไปที่ RADIUS Server เพื่อขอตรวจสอบสิทธิ์ (Request Authentication)

3. RADIUS Server ทำการตรวจสอบสิทธิ์โดยใช้ข้อมูลที่ NAS ส่งมา (Access-Request) กับข้อมูลที่จัดเก็บไว้ใน RADIUS Server เอง หรือจากฐานข้อมูลภายนอก อื่น ๆ เช่น MS SQL Server, Oracle Database, LDAP Database หรือ RADIUS Server อื่น (ซึ่งเรียกการส่งผ่านการตรวจสอบสิทธิ์แบบนี้ว่า Proxy) ในกรณีที่ข้อมูลทั้งหมดถูกต้อง RADIUS Server จะส่งผลยินยอมการเชื่อมต่อ (Access-Accept) หรือ ไม่ยินยอม (Access-Reject) ในกรณีที่ข้อมูลไม่ถูกต้อง แก่ NAS หลังจากนั้น NAS จะเชื่อมต่อหรือยกเลิกการการต่อเชื่อมตามผลที่ได้รับจาก RADIUS Server ซึ่งตามปรกติแล้ว NAS จะขอบันทึกข้อมูลต่าง ๆ เช่น วันที่ เวลา Username และข้อมูลอื่น ๆ ไปที่ RADIUS Server (Accounting Request) เพื่อให้ RADIUS Sever จัดเก็บข้อมูลหรือส่งต่อไปที่ RADIUS Server อื่น จัดเก็บเพื่อใช้ในการประมวลผลอื่น ๆ ต่อไป

RADIUS Package คือ ข้อมูลที่ถูกส่งหรือรับระหว่าง RADIUS Server และ RADIUS Client (หมายถึง NAS) มีรูปแบบที่ถูกกำหนดไว้ตามมาตรฐานของ RFC 2685 Remote Authentication Dial In User Service (RADIUS) และ 2866 RADIUS Accounting



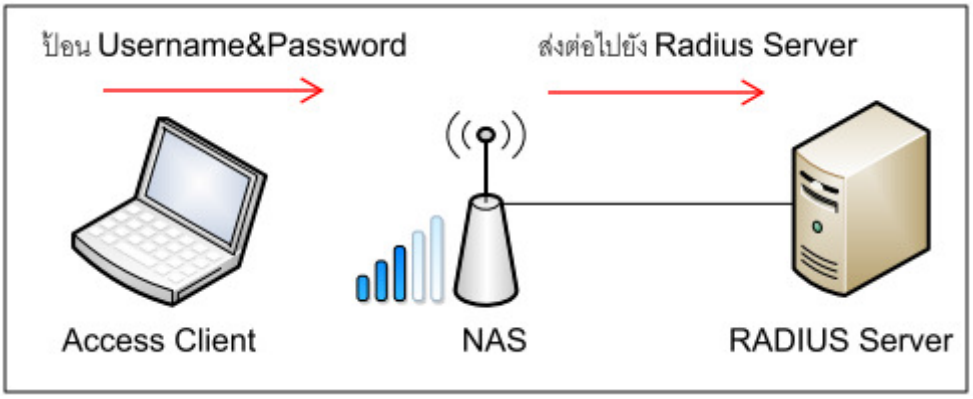
ภาพที่ 2.2 แสดงองค์ประกอบพื้นฐานของ RADIUS Server

กระบวนการการทำงานของ Freeradius เริ่มแรกหลังจากที่ได้ มีการสร้างแอคเคาท์เป็นที่เรียบร้อยแล้ว มีการใช้งานโปรแกรมradius-client เพื่อล็อกอินหรือตรวจสอบสิทธิ ก็จะเข้าสู่กระบวนการต่าง ๆ ดังนี้

1) โปรแกรม Radius-Client จะติดต่อโปรแกรม Freeradius ตามหมายเลขไอพีและพอร์ตที่ได้กำหนดไว้ โดยปรกติพอร์ทของโปรแกรมจะอยู่ที่ 1812 ตาม Default

2) โปรแกรมจะนำชื่อแอคเคาท์ รหัสผ่าน และค่า Shared Secret ไปตรวจสอบว่าถูกต้องหรือไม่ในขั้นตอนนี้จะมีกระบวนการดังนี้

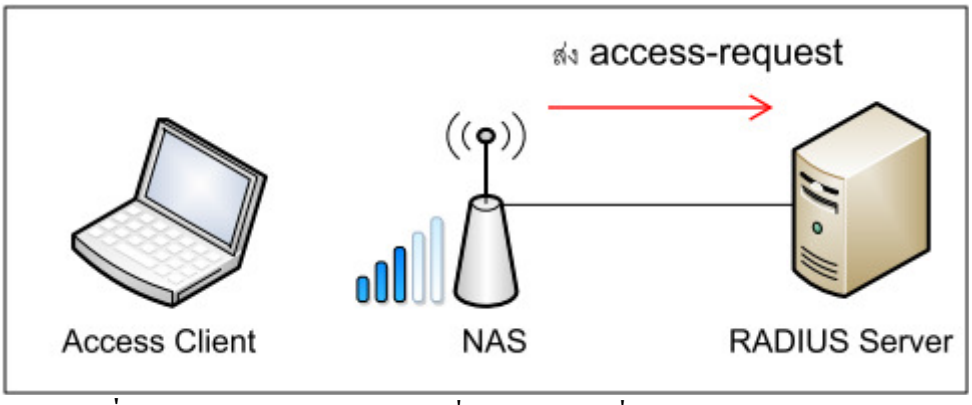
2.1) Radius-Client ป้อน {Username, Password, Secret key} ส่งต่อไปยัง Freeradius



ภาพที่ 2.3 กระบวนการทำงานของเมื่อร้องขอการเชื่อมต่อกับ Radius Server 1

ในขั้นตอนนี้จะมีการส่งยูสเซอร์เนม พาสเวิร์ด และ ซีเคร็ดคีย์ไปยังเซิร์ฟเวอร์เพื่อใช้ ในการตรวจสอบความถูกต้อง

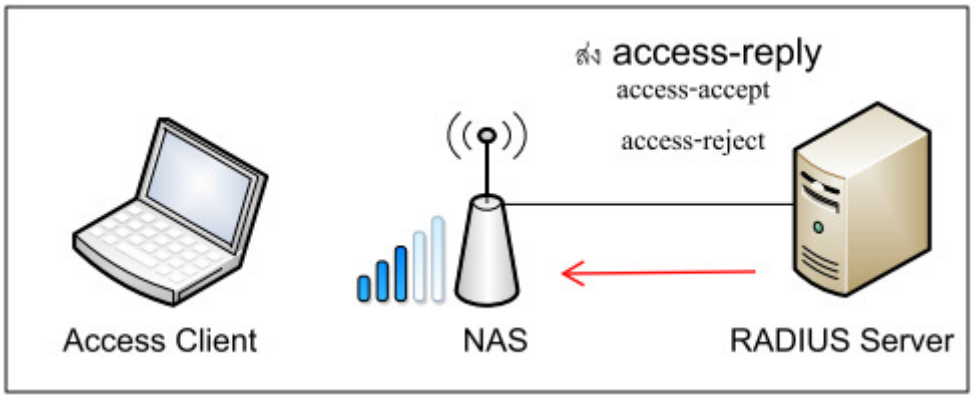
2.2) Radius-Client ส่ง {Access-Request} ส่งไปยัง Freeradius



ภาพที่ 2.4 กระบวนการทำงานของเมื่อร้องขอการเชื่อมต่อกับ Radius Server 2

ในขั้นตอนนี้ ทางฝั่งไคลเอ้นท์จะสร้างสัญญาณร้องขอผลตอบกลับมาจากเซิร์ฟเวอร์หรือรอสัญญาณตอบรับความถูกผิดของข้อมูลที่ส่งจากขั้นตอนแรก

2.3) Freeradius ส่ง {Access-Reply} ส่งต่อไปยัง Radius-Client



ภาพที่ 2.5 กระบวนการทำงานของเมื่อร้องขอการเชื่อมต่อกับ Radius Server 3

ในขั้นตอนนี้เซิร์ฟเวอร์จะตอบกลับไปยังเครื่องไคลเอ้นท์ด้วยสัญญาณ Accessreply โดยสัญญาณนี้จะประกอบไปด้วย2สัญญาณย่อยที่สำคัญแต่จะเกิดขึ้นเพียงแค่หนึ่ง สัญญาณต่อเงื่อนไขนั้น คือสัญญาณ Access-Accept และ Access-Reject โดยสัญญาณ Access-Accept นี้คือสัญญาณที่ใช้ตอบกลับไปยังไคลเอ้นท์ว่ายูสเซอร์เนม รหัสผ่านและซีเคร็ดคีย์นั้นถูกต้องส่วน สัญญาณ Access-Reject นั้นจะตรงกันข้ามกับสัญญาณแรก โดยมีความหมายคือยูสเซอร์เนม รหัสผ่าน และซีเคร็ดคีย์ไม่ถูกต้องหรืออาจมี เฉพาะตัวหนึ่งตัวใดไม่ถูกต้องก็ได้ เป็นต้น ในโปรแกรม Freeradius ต้องอาศัยฐานข้อมูลเพื่อดึงข้อมูลมาใช้ในการประมวลผลไม่ว่าจะเป็นยูสเซอร์เนม และพาสเวิร์ดหรือเมสเสจและเงื่อนไขต่างๆของแต่ละยูสเซอร์ โดยในฐานข้อมูลจะมีตารางที่เกี่ยวข้องดังนี้ Radcheck, Radgroupcheck, Radgroupreply, Usergroup และ Radacct ซึ่งตารางดังกล่าวก็ได้บอกคุณสมบัติไปแล้ว

RADIUS Port

RADIUS Server จำเป็นต้องระบุ UDP Port เพื่อใช้สำหรับรับและส่ง Authentication และ Accounting Package ระหว่าง RADIUS Server และ RADIUS Client ซึ่งเริ่มต้นที่ RADIUS ได้ถูกพัฒนาขึ้นผู้พัฒนาได้ใช้ Port 1645 สำหรับการส่งและรับ Package Authentication และ 1646 สำหรับการส่งและรับ Package Accounting แต่เนื่องจากมาตรฐานนั้นได้มีการกำหนด Port ดังกล่าวสำหรับ “Datametrics” ดังนั้น Port ที่เป็นมาตรฐานในปัจจุบันนี้ คือ 1812 สำหรับการส่งและรับ Package Authentication, 1813 สำหรับการส่งและรับ Package Accounting

Password Protocols เนื่องจากการส่ง Access-Request ในขณะที่มีการขอ Authentication มีการส่ง Password จาก NAS ไปยัง RADIUS Server จึงจำเป็นต้องคำนึงถึงความปลอดภัยของ Password ดังกล่าว ดังนั้นจึงมีการสร้างโพรโตคอลสำหรับใช้งานในส่วนนี้ขึ้นซึ่งได้แก่

PAP (Password Authentication Protocol) ในขณะที่มีการขอเชื่อมต่อ(User Negotiates) จาก Access Clients มายัง NAS การส่ง Password ในขั้นตอนนี้จะยังไม่มีการเข้ารหัส (Encrypt) ใดๆ Password จะจัดส่งในรูปแบบ “Clear Text” เมื่อ NAS รวบรวมข้อมูลที่เพียงพอสำหรับสร้าง Access-Request แล้ว NAS จะ Encrypt Password โดยใช้ Authentication Shared Secret ที่ถูกกำหนดไว้ แล้วส่ง Access-Request ดังกล่าวไปยัง RADIUS Server เมื่อ RADIUS Server ได้รับ Access-Request จาก NAS แล้วจะทำการ Decrypt Password ที่ได้รับโดยใช้ Authentication Shared Secret ที่จัดเก็บไว้สำหรับ NAS ตัวดังกล่าว โพรโตคอล PAP สามารถใช้ได้กับ RADIUS Server ทุกตัว โดย CHAP (Challenge Handshake Authentication Protocol) สำหรับ CHAP ได้ถูกสร้างขึ้นเพื่อหลีกเลี่ยงการส่ง Password แบบ “Clear Text” ในขณะที่ User Negotiates เมื่อ NAS รับทราบแล้ว NAS จะสร้าง Challenge โดยสุ่มตัวอักษร แล้วส่งกลับไปยัง Access Client เมื่อ Access Client ได้รับ Challenge จะทำการสร้าง Digest คือ นำ Challenge ที่ได้รับมาต่อท้าย Password แล้วทำการ Encrypt แบบ one-way Encryption (MD5 Algorithm) แล้วส่ง Digest นั้นแทน Password ไปยัง NAS และ NAS จะสร้าง Access-Request สำหรับการ Authentication และส่งไปยัง RADIUS Server เนื่องจาก Digest ถูกสร้างแบบ One-Way Encryption ไม่สามารถ Decrypt ได้ RADIUS Server จึงจำเป็นต้องใช้ Attribute ที่เกี่ยวกับ CHAP Protocol ที่ถูกจัดส่งมาใน Access-Request Package ที่ได้รับจาก NAS ซึ่งมี 2 Attributes ที่เกี่ยวข้องดังนี้

- CHAP-Password : Attribute สำหรับ Digest (Password ที่ต่อท้ายด้วย Challenge แล้ว Encrypt ด้วย MD5 Algorithm)

- CHAP-Challenge : Attribute สำหรับ Challenge ที่ถูกสุ่มขึ้นโดย NAS RADIUS Server ใช้ Challenge จาก CHAP-Challenge ต่อท้าย Password ที่จัดเก็บไว้นำมา Encrypt ด้วยวิธี MD5 แล้วเปรียบเทียบกับ CHAP-Password ที่ได้รับ

MS-CHAP และ MS-CHAP-V2

MS-CHAP (Microsoft Challenge Handshake Authentication Protocol) ทั้ง 2 เวอร์ชั่นของ MS-CHAP ใช้วิธีการของโพรโตคอล CHAP แต่มีส่วนเพิ่มเติมขึ้นโดย Microsoft ข้อมูลเพิ่มเติมให้ดูที่ RFC 2433 2548 และ 2759.

สรุปขั้นตอนการ Authentication

วิธีและการกำหนดลำดับการ Authenticate (Authentication Method) Native User Authentication คือการตรวจสอบ Username Password หรือ ข้อมูลอื่น ๆ จากข้อมูลที่ RADIUS Server จัดเก็บไว้ที่ตัวเอง ซึ่งเราเรียกสั้นๆ ว่า Native User

Pass-Through Authentication คือการส่งผ่านการ Authenticate ไปยังระบบการตรวจสอบอื่น ๆ เช่น Windows NT Database, Active Directory ใน Windows 2000, ACE/Server (SecurID) หรือ TACACS + Server

Proxy RADIUS Authentication คือการส่งผ่านการ Authenticate ไปยัง RADIUS Server ตัวอื่นเพื่อทำหน้าที่ตรวจสอบแทน และส่ง Access-Accept หรือ Access-Reject กลับมาที่ RADIUS Server ตัวเดิม เพื่อจัดส่งให้กับ NAS ต่อไป

External Authentication คือการตรวจสอบที่เป็นการทำงานร่วมกันระหว่าง RADIUS Server กับฐานข้อมูลต่าง ๆ เช่น Microsoft SQL, Oracle Database หรือ LDAP Server Database RADIUS Server จะขอข้อมูลที่ต้องการ เช่น Username, Password จากฐานข้อมูลแล้วนำมาเปรียบเทียบกับ Access-Request

Authenticate-Only Request เราสามารถกำหนดให้ RADIUS Server แจ้งเฉพาะผลการ Authenticate เท่านั้นใน Access-Accept หรือ Access-Reject โดยการกำหนดค่า Service-Type ที่ NAS เป็น AuthenticateOnly (Cool) นอกจาก RADIUS Sever สามารถ Authenticate ได้หลายวิธีตามที่กล่าวข้างต้นแล้ว เรายังสามารถกำหนดลำดับ Authentication Method ดังกล่าวให้ทำงานร่วมกันได้ด้วย เช่น กำหนดให้ RADIUS Server Authenticate ตามลำดับขั้นดังนี้ Native User, External 1 (SQL Database), External 2 (Oracle Database)

การ Authenticate จะมีขั้นตอนดังนี้ RADIUS Server จะตรวจสอบที่ Native User ก่อนในกรณีที่ไม่พบหรือไม่ถูกต้องจะเลื่อนไปตรวจสอบที่ SQL Database และ Oracle Database ตามลำดับ ซึ่ง RADIUS Server จะยังไม่ส่ง Access-Reject จนกว่าจะทำจนครบทุก Method ที่กำหนดไว้

แต่ในกรณีที่ถูกต้องตามเงื่อนไขที่กำหนด RADIUS Server จะส่ง Access-Accept ไปที่ NAS ทันทีโดยไม่ต้องตรวจสอบจนครบทุก Method Directed Authentication คือ การกำหนดให้ RADIUS Server ข้าม Authenticate Method ที่ได้ถูกกำหนดไว้ที่ Authenticate Method List ไปยัง Authenticate Method ที่ระบุเลย โดยไม่ต้องตรวจสอบตามลำดับที่กำหนดไว้ เราสามารถใช้งาน Directed Authentication โดยการกำหนด Realm ขึ้นเพื่อใช้ตรวจสอบ

TCPDUMP

Tcpdump เป็นโปรแกรมประเภทเดียวกับ Sniffer, Wireshark คือใช้ในการดักจับ (capture) Traffic หรือ Packet ที่ รับ/ส่ง เข้า/ออก ระหว่างพอร์ตแลน (LAN) ของเซิรฟ์เวอร์เครื่องที่รันคำสั่ง และอุปกรณ์เครือข่าย (Router, Switch, HUB) มีประโยชน์อย่างมาก เพื่อใช้ในการวิเคราะห์ ตรวจสอบ หรือแก้ปัญหาเกี่ยวกับ Network ได้ Tcpdump ต้องรันด้วย root หรือเทียบเท่า และรันแบบ command line ติดตั้งมาเป็นดีฟอลต์บนลีนุกซ์เกือบทุกตระกูล เวอร์ชั่น จึงใช้งานได้สะดวก ไม่ต้องติดตั้งเพิ่มเติมเหมือนโปรแกรมอื่นๆ ในที่นี้ขอแนะนำวิธีการใช้งานเบื้องต้นของ Tcpdump

ผลลัพธ์ที่แสดงจากคำสั่ง tcpdump จะแตกต่างกันไปขึ้นกับ Network Protocol ที่รับ/ส่ง เช่นถ้าเป็น IP Protocol (TCP, UDP) รูปแบบจะเป็น เวลา, IP, [Source IP Address].[Source Port], [Destination Address].[Destination Port], IP, TCP, UDP Headers เลือกพอร์ตที่ดักจับ

ระบุออปชั่น “-i” แล้วตามด้วยชื่อพอร์ต เช่น ต้องการจับ Packet ที่เข้าออก eth1

```
[root@server ~]# tcpdump -i eth1
tcpdump: WARNING: eth1: no IPv4 address assigned
tcpdump: verbose output suppressed, use -v or -vv for full
protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size
65535 bytes
^C
```

ภาพที่ 2.6 คำสั่ง tcpdump -i eth1

ในที่นี้ พอร์ต eth1 ไม่มีการใส่ IP Address จึงขึ้นฟ้อง “WARNING: eth1: no IPv4 address assigned” แต่ก็ยังสามารถดักจับ Traffic ได้ไม่ต้องแปลง IP เป็นชื่อ hostname

ผลลัพธ์ที่แสดงออกมา โปรแกรม tcpdump จะพยายามแปลง IP Address ทั้งต้นทาง ปลายทาง ของ packet ที่ดักจับได้ ให้เป็นชื่อ hostname โดยใช้ไฟล์ /etc/hosts หรือ บริการ DNS

ถ้าหาก traffic มีปริมาณมาก แนะนำให้ปิดคุณสมบัติการพยายามแปลง IP เป็น ชื่อ hostname ออกซะ ด้วยการระบุออปชั่น “-n” เพื่อลดโหลด DNS

```
[root@server ~]# tcpdump -i eth0 -l -n
tcpdump: verbose output suppressed, use -v or -vv for full
protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size
65535 bytes
```

```
09:31:10.322145 IP 192.168.1.1.ssh > 192.168.1.102.jwalkserver:
Flags [P.], seq 1901277915:1901278111, ack 2244887520, win 8576,
length 196
09:31:10.322752 IP 192.168.1.102.jwalkserver > 192.168.1.1.ssh:
Flags [.], ack 196, win 65535, length 0
09:31:10.330843 IP 192.168.1.1.ssh > 192.168.1.102.jwalkserver:
Flags [P.], seq 196:472, ack 1, win 8576, length 276
09:31:10.332785 IP 192.168.1.1.ssh > 192.168.1.102.jwalkserver:
Flags [P.], seq 472:636, ack 1, win 8576, length 164
09:31:10.333209 IP 192.168.1.102.jwalkserver > 192.168.1.1.ssh:
Flags [.], ack 636, win 65095, length 0
^C
```

ภาพที่ 2.7 คำสั่ง tcpdump -i eth0 -l –n ไม่ต้องแปลง Port Number เป็นชื่อ Port Name

หากต้องการแสดงชื่อพอร์ต (TCP, UDP ports) เป็นตัวเลข (ไฟล์ /etc/services) ให้ระบุออปชั่น “-n” เพิ่มอีกหนึ่ง หรือระบุเป็น “-nn” เลย

```
[root@server ~]# tcpdump -i eth0 -l -nn
tcpdump: verbose output suppressed, use -v or -vv for full
protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size
65535 bytes
09:39:38.800430 IP 192.168.1.102.1289 > 192.168.1.1.22: Flags
[.], ack 196, win 65535, length 0
09:39:38.802793 IP 192.168.1.1.22 > 192.168.1.102.1289: Flags
[P.], seq 196:456, ack 1, win 8576, length 260
09:39:38.805709 IP 192.168.1.1.22 > 192.168.1.102.1289: Flags
[P.], seq 456:604, ack 1, win 8576, length 148
09:39:38.805963 IP 192.168.1.102.1289 > 192.168.1.1.22: Flags
[.], ack 604, win 65127, length 0
09:39:38.808655 IP 192.168.1.1.22 > 192.168.1.102.1289: Flags
[P.], seq 604:848, ack 1, win 8576, length 244
^C
```

ภาพที่ 2.8 คำสั่ง tcpdump -i eth0 -l -nn

หากไม่สนใจ ข้อมูลใน IP, TCP, UDP Headers ต้องการรู้แค่ว่า IP อะไรคุยกัน ใช้พอร์ตอะไร แค่นั้นพอ ให้ระบุออปชั่น “-q”

```
[root@server ~]# tcpdump -i eth0 -l -nn -q
```

```
tcpdump: verbose output suppressed, use -v or -vv for full
protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size
65535 bytes
09:40:48.472427 IP 192.168.1.1.22 > 192.168.1.102.1289: tcp 196
09:40:48.473006 IP 192.168.1.102.1289 > 192.168.1.1.22: tcp 0
09:40:48.479252 IP 192.168.1.1.22 > 192.168.1.102.1289: tcp 164
09:40:48.487003 IP 192.168.1.1.22 > 192.168.1.102.1289: tcp 100
09:40:48.487311 IP 192.168.1.102.1289 > 192.168.1.1.22: tcp 0
^C
```

ภาพที่ 2.9 คำสั่ง tcpdump -i eth0 -l -nn -q

แสดงข้อมูล Layer 2 (MAC Address) ระบุออปชั่น “-e” หากต้องการแสดงข้อมูล Layer 2 หรือแสดง Source, Destination MAC Address ด้วย

```
[root@server ~]# tcpdump -i eth0 -l -nn -e -q
tcpdump: verbose output suppressed, use -v or -vv for full
protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size
65535 bytes
09:44:05.644395 00:11:22:33:44:55 > 00:55:66:77:88:99, IPv4,
length 250: 192.168.1.1.22 > 192.168.1.102.1289: tcp 196
09:44:05.644959 00:55:66:77:88:99 > 00:11:22:33:44:55, IPv4,
length 60: 192.168.1.102.1289 > 192.168.1.1.22: tcp 0
09:44:05.652121 00:11:22:33:44:55 > 00:55:66:77:88:99, IPv4,
length 314: 192.168.1.1.22 > 192.168.1.102.1289: tcp 260
09:44:05.660897 00:11:22:33:44:55 > 00:55:66:77:88:99, IPv4,
length 218: 192.168.1.1.22 > 192.168.1.102.1289: tcp 164
09:44:05.661173 00:55:66:77:88:99 > 00:11:22:33:44:55, IPv4,
length 60: 192.168.1.102.1289 > 192.168.1.1.22: tcp 0
^C
```

ภาพที่ 2.10 คำสั่ง tcpdump -i eth0 -l -nn -e -q

หากต้องการบันทึก (save) ผลลัพธ์ที่ดักจับได้ ให้เหมือนกับหน้าจอที่แสดงขึ้นมา ก็ใช้การ Redirection ตัวอย่าง การเก็บผลลัพธ์ลงไฟล์ ให้เหมือนกับที่แสดงขึ้นมา

```
[root@server ~]# tcpdump –i eth0 –l –nn –q > capture-display.log
```

```
tcpdump: verbose output suppressed, use -v or -vv for full
protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size
65535 bytes
^C
7 packets captured
7 packets received by filter
0 packets dropped by kernel
```

ภาพที่ 2.11 คำสั่ง tcpdump -i eth0 -l -nn -q > capture-display.log

ใช้คำสั่ง cat เพื่อดูเนื้อหาไฟล์ที่บันทึก

```
[root@server ~]# cat capture-display.log
09:57:08.118314 IP 192.168.1.1.22 > 192.168.1.102.1289: tcp 196
09:57:08.118515 IP 192.168.1.102.1289 > 192.168.1.1.22: tcp 0
09:57:09.138258 IP 192.168.1.102.1289 > 192.168.1.1.22: tcp 36
09:57:09.355761 IP 192.168.1.102.1289 > 192.168.1.1.22: tcp 36
09:57:09.356025 IP 192.168.1.1.22 > 192.168.1.102.1289: tcp 0
^C
```

ภาพที่ 2.12 คำสั่ง cat capture-display.log

แต่ถ้าต้องการบันทึกแบบเต็มรูปแบบ ให้ใช้ออปชั่น “-w” แล้วตามด้วยชื่อไฟล์ โดยทั่วไปนิยมบันทึกเป็นไฟล์นามสกุล “.cap”

```
[root@server ~]# tcpdump -i eth0 -w capture-raw.cap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture
size 65535 bytes
^C
24 packets captured
24 packets received by filter
0 packets dropped by kernel
```

ภาพที่ 2.13 คำสั่ง tcpdump -i eth0 -w capture-raw.cap

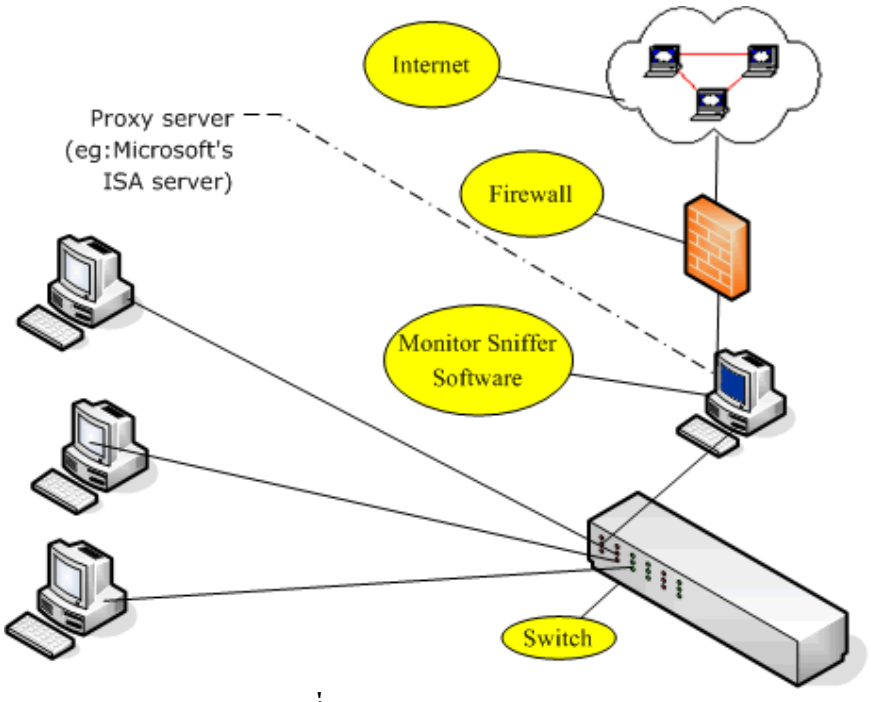
ไฟล์ที่บันทึกด้วยออปชั่น “-w” สามารถนำมาเปิดย้อนหลังด้วยคำสั่ง tcpdump ตามด้วยออปชั่น “-r” นอกจากนี้ ยังสามารถนำไปเปิดกับโปรแกรม Wireshark ได้อีกด้วย

```
[root@server ~]# tcpdump -r capture-raw.cap -l -nn
reading from file capture-raw.cap, link-type EN10MB (Ethernet)
10:00:14.959339 IP 192.168.1.102.1289 > 192.168.1.1.22: Flags
[P.], seq 105:157, ack 236, win 65431, length 52
10:00:14.972070 IP 192.168.1.1.22 > 192.168.1.102.1289: Flags
[P.], seq 236:288, ack 157, win 8576, length 52
10:00:15.137079 IP 192.168.1.102.1289 > 192.168.1.1.22: Flags
[.], ack 288, win 65379, length 0
10:00:15.221339 IP 192.168.1.102.1289 > 192.168.1.1.22: Flags
[P.], seq 157:209, ack 288, win 65379, length 52
10:00:15.227905 IP 192.168.1.1.22 > 192.168.1.102.1289: Flags
[P.], seq 288:340, ack 209, win 8576, length 52
^C
```

ภาพที่ 2.14 คำสั่ง tcpdump -r capture-raw.cap -l -nn

Sniffer

Sniffer คือโปรแกรมที่เอาไว้ดักจับข้อมูล บนระบบ Network เนื่องจากคอมพิวเตอร์เน็ตเวิร์คเป็นระบบการสื่อสารที่ใช้ร่วมกัน เพื่อประหยัดค่าใช้จ่าย การแบ่งกันใช้ (Sharing) หมายถึง คอมพิวเตอร์สามารถรับข้อมูลที่คอมพิวเตอร์เครื่องอื่นตั้งใจจะส่งไป ให้อีกเครื่องหนึ่ง การดักจับข้อมูลที่ผ่านไปมาระหว่าง เน็ตเวิร์คเรียกว่า Sniffing (คล้ายๆ การดักฟังโทรศัพท์ แต่การดักฟังโทรศัพท์จะทำได้ทีละเครื่อง แต่ Sniffer ทำได้ทีเดียวทั้ง Network เลย)



ภาพที่ 2.15 ตำแหน่งของ Sniffer

แรกเริ่ม Sniffer เป็นชื่อโปรแกรมของบริษัท Network Associates Inc. สหรัฐฯ เพื่อใช้ในผลิตภันณฑ์ของตนเองในเครือ Sniffer Network Analyzer ซึ่งเป็นโปรแกรมวิเคราะห์ Network โดยอาศัยการดักอ่านข้อมูล แรกเริ่ม มันถูกสร้างมาเพื่อการป้องกัน (คือเอามาตรวจสิ่งที่วิ่งอยู่บน Network นั่นเอง) แต่เมื่อมี Hacker หัวใส นำโปรแกรมที่ใช้ป้องกัน ไปทำลาย มันจึงเป็นดาบ2คม คือ หากเราเอามาใช้ในการตรวจ ระบบ Network ของเรา ก็เป็นประโยชน์ (แม้บางครั้งอาจจะละเมิดความเป็นส่วนตัวไปบ้าง) แต่หากเราเอาไปใช้ในการดักอ่าน ของข้อมูล เพื่อ Hack ล่ะ เช่น เอาไปวางไว้บนทางของ ระบบ E-Mail ก็ทำให้ข้อมูลทั้งหมด ของ E-Mail โดนอ่าน (ทั้งๆ ที่มันควรจะเป็นความลับ)

การป้องกันการถูกดักอ่านข้อมูลโดย Sniffer

1. อย่างแรกเลย เปลี่ยนจาก Hub มาใช้ Switch

2. หลีกเลี่ยงการส่งข้อมูลที่ไม่มีการเข้ารหัส

3. ให้ตระหนักว่า ใน Network นั้นสามารถถูกดักอ่านได้เสมอ เพราะฉะนั้นการส่งข้อมูลแต่ละครั้ง ต้องประเมินว่า หากโดนดักอ่านแล้วจะคุ้มกันไหม หากมีความสำคัญมากควรหาวิธีอื่นในการส่งข้อมูล

4. หากมีการใช้บริการเกี่ยวกับด้านการเงิน หรือข้อมูลรหัสผ่าน ให้เลือกใช้ผู้บริการที่เข้ารหัสข้อมูลด้วย SSL

5. หากสามารถเพิ่มความปลอดภัยของการส่งข้อมูลด้วยการเข้ารหัส ก็จะเป็นวิธีที่ดี แม้การส่งแบบนี้จะโดนดักได้ แต่ข้อมูลมีการเข้ารหัสไว้ ทำให้คนที่ดักไป ต้องไปนั่งปวดหัวถอดกันอีก โดยใช้โปรแกรมเข้ารหัสไฟล์

6. หากมีการสื่อสารข้อมูลภายในองค์กรโดยผ่านอินเทอร์เน็ต การนำเทคโนโลยีของ VPN (Vitual Private Network) มาใช้จะช่วยเพิ่มความปลอดภัยได้

การใช้ประโยชน์จาก Sniffer

1. Network Analyzer นั้นคือใช้ประเมิน Network ว่ามี Packet (หรือข้อมูล) ที่วิ่งไปวิ่งมานั้น มีอะไรบ้าง และ แพ็กเก็ต ที่วิ่งไปวิ่งมา มีอันตรายอะไรหรือเปล่า มีผู้ใช้มาน้อยเพียงไร เวลาใดมีคนใช้เยอะและเวลาใดมีคนใช้น้อย ผู้ใช้ ใช้แบนด์วิดธ์ไปในทางไหนบ้าง โดยสามารถเอาข้อมูลเหล่านี้มาประเมินเพื่อจัดการระบบ network ของเราได้

2. Network Debugging Tools ใช้ตรวจสอบข้อผิดพลาดใน Network เพื่อจะดูว่า การส่งข้อมูลนั้นถูกต้องหรือไม่ มีอะไรแปลกปลอมวิ่งอยู่รึเปล่า โดนเฉพาะกรณีที่มีการใช้เครื่องมือระดับ Network มาเกี่ยวด้วย เช่น ส่งไฟล์ผ่าน fire wall แล้วมีปัญหา หรือการทดสอบ ACL (Access Control List) ของเราเตอร์ เป็นต้น หากไม่มี Sniffer แล้วเราก็จะหาต้นตอของปัญหาได้ยาก

3. Packet Monitoring ใช้ในกรณีการศึกษาโปรโตคอลในระดับ Network จำเป็นต้องเห็นข้อมูลที่มันสื่อสารกันจึงจะเห็นภาพจริงได้ Packet Monitoring เป็นการนำแพ็กเก็ตมาแสดงให้ดูให้ผู้ใช้เห็นในรูปแบบต่างๆ เช่นการ Scan ของ Hacker หากไม่มีเครื่องมือประเภท Sniffer แล้วเราก็จะรู้ได้ลำบาก

4. IDS (Intrusion Detection System) ใช้ตรวจจับผู้บุกรุก หากมีข้อมูลที่เป็นอันตรายตามที่มันได้ถูก Config ไว้มันก็จะเตะข้อมูลหรือแพ็กเก็ตนั้นทิ้งไป และหากมันพบว่าข้อมูลไม่เป็นอันตราย มันก็จะอนุญาตให้ผ่านไป

MySQL

MySQL จัดเป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS : Relational Database Management System) ซึ่งเป็นที่นิยมใช้กันมากในปัจจุบัน โดยเฉพาะอย่างยิ่งในโลกของ internet เนื่องจาก

- MySQL เป็นฟรีแวร์ทางด้านฐานข้อมูลที่มีประสิทธิภาพสูง
- นักพัฒนาฐานข้อมูลที่เคยใช้ MySQL ต่างยอมรับในความรวดเร็ว การรองรับจำนวนผู้ใช้ และขนาดของข้อมูลจำนวนมหาศาล

- สนับสนุนการใช้งานบนระบบปฏิบัติการมากมาย เช่น UNIX OS/2 MAC OS Windows
- สามารถใช้งานร่วมกับ Web Development platform เช่น C, C++ , Java, Perl, PHP, Python, TCL, หรือ ASP
- ได้รับความนิยมอย่างมากในปัจจุบัน และมีแนวโน้มสูงขึ้นเรื่อยๆ ในอนาคต

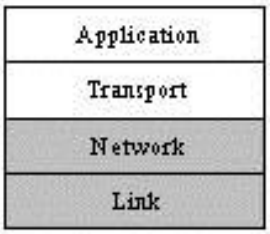
MySQL จัดเป็นซอฟท์แวร์ประเภท Open Source Software สามารถ Download ซอร์สโคดต้นฉบับได้จากอินเทอร์เนทโดยไม่เสียค่าใช้จ่ายใดๆ การแก้ไขสามารถทำได้ตามต้องการ MySQL ยึดถือสิทธิบัตรตาม GPL (GNU General Public License) ซึ่งเป็นข้อกำหนดของซอฟท์แวร์ประเภทนี้ โดยจะเป็นการชี้แจงว่าสิ่งใดทำได้ หรือทำไม่ได้ในกรณีต่างๆ สามารถหาข้อมูลเพิ่มเติมได้จากเว็บไซท์ www.gnu.org

ทุกวันนี้มีการนำ MySQL ไปใช้ในระบบต่างๆมากมาย ไม่ว่าจะเป็นระบบเล็กๆที่มีจำนวนตารางข้อมูลน้อย เช่น ระบบฐานข้อมูลของแผนกเล็กๆ ไปจนถึงระบบฐานข้อมูลขนาดใหญ่ เช่น ระบบบัญชีเงินเดือน ในปัจจุบันได้มีการใช้ MySQL เป็น Database Server เพื่อการทำงานสำหรับฐานข้อมูลบนเว็บมากขึ้น

ความรู้เบื้องต้น เกี่ยวกับ TCP/IP

Transmission Control Protocol/Internet Protocol โปรโตคอล TCP/IP เป็นชุดของโปรโตคอลที่มีการพัฒนามาตั้งแต่ปี 1960 โดยมีวัตถุประสงค์ให้สามารถใช้สื่อสารจากต้นทางข้ามเน็ตเวิร์กไปยังปลายทางได้ และสามารถหาเส้นทางที่จะส่งข้อมูลไปได้เองโดยอัตโนมัติ ถึงแม้ว่าในระหว่างทางอาจจะผ่านเน็ตเวิร์กที่มีปัญหาโปรโตคอลที่ยังคงหาเส้นทางส่งผ่านข้อมูลไปให้ถึงปลายทางได้ ในระยะเริ่มต้นโปรโตคอลนี้ใช้กันในวงแคบ ๆ เฉพาะราชการและสถานศึกษาของอเมริกาจนในช่วงปี 90 จึงมีการนำมาใช้ในทางธุรกิจและเป็นจุดเริ่มต้นของอินเทอร์เน็ตในปัจจุบันผมคิดว่าประวัติโดยส่วนใหญ่ของโปรโตคอลนี้คงมีอยู่ในหนังสือที่เกี่ยวข้องกับอินเทอร์เน็ตจำนวนหลายเล่มแล้วก็คงอนุญาตละไว้เพื่อให้เข้าสู่จุดมุ่งหมายของหนังสือฉบับนี้ได้อย่างรวดเร็วจากที่กล่าวมาข้างต้นจะเห็นได้ว่า TCP/IP นี้มีการออกแบบมาเป็นเวลานาน มาตั้งแต่ปี 1960 ก็มีการใช้ไปปรับปรุงอยู่เรื่อย ๆ เพื่อให้สามารถใช้งานได้หลากหลายและมีประสิทธิภาพมากขึ้นแต่อย่างไรก็ตามโปรโตคอลนี้ก็ยังคงมีจุดบกพร่องอีกมากมายซึ่งจุดบกพร่องเหล่านี้บางส่วนก็มิได้มีผลกระทบรุนแรงเท่าไหร่นักในมุมมองของนักคอมพิวเตอร์โดยทั่วไปแต่กลับกลายเป็นเครื่องมืออันทรงอานุภาพของบรรดาเหล่าแฮกเกอร์ทั้งหลายที่หยิบฉวยข้อบกพร่องเหล่านี้มาใช้ในการโจมตีผู้อื่นโดยเฉพาะการโจมตีแบบ Dos นั้นล้วนแต่ใช้ข้อบกพร่องของ TCP/IP แทบทั้งสิ้น

การศึกษาโปรโตคอลนี้ตามปกติที่ทำกันทั่วไปอาจไม่เพียงพอที่จะทำให้รู้เท่าทันกลวิธีของแฮกเกอร์และป้องกันตนเองได้ดังนั้นสิ่งที่เราจำเป็นต้องศึกษาควบคู่กันไปกับวิธีการใช้งานโปรโตคอลก็คือข้อบกพร่องของโปรโตคอลคืออะไรมีผลกระทบอย่างไรและสามารถป้องกันตัวเองได้อย่างไร อย่างไรก็ตามก่อนที่จะศึกษาเรื่อง IDS เราจำเป็นต้องเริ่มศึกษาโปรโตคอล TCP/IP อย่างละเอียดถี่ถ้วยเสียก่อนเพื่อเป็นพื้นฐานในการทำความเข้าใจเนื้อหาที่มีความซับซ้อนในภายหลังในการแบ่งชั้น Layering TCP/IP เป็นชุดของโปรโตคอลที่ประกอบด้วยโปรโตคอลย่อยหลายตัวโดยแต่ละตัวจะทำหน้าที่ในแต่ละชั้นหรือเลเยอร์ layer ซึ่งรับผิดชอบและแปลความหมายของข้อมูลในแต่ละระดับของการสื่อสาร ซึ่งในภาพรวมแล้ว TCP/IPแบ่งออกเป็น 4 เลเยอร์ ดังรูป



ภาพที่ 2.16 TCP/IP Layer

Link Layer

ในเลเยอร์นี้จะเป็นดีไวซ์ไดรเวอร์ที่ทำงานอยู่บนระบบปฏิบัติการแต่ละระบบทำหน้าที่รับผิดชอบในการรับส่งสัญญาณไฟฟ้าจนเป็นข้อมูลทางคอมพิวเตอร์ โปรโตคอลระดับนี้ เช่น Ethernet และ SLIP (Serial Line Internet Protocol) Network Layer รับผิดชอบในการรับส่งข้อมูลในเน็ตเวิร์กส่งต่อข้อมูลไปจนถึงจุดหมายปลายทางโปรโตคอลระดับนี้ได้แก่ IP, ICMP, IGMP Transport Layer รับผิดชอบในการรับส่งข้อมูลระหว่างเครื่องหนึ่ง (Host) ไปยังอักโฮสต์หหนึ่งจะส่งข้อมูลไปให้ Application Layer นำไปใช้งานต่อ มีโปรโตคอลที่จัดอยู่ในเลเยอร์นี้ คือ TCP และ UDPซึ่งมีลักษณะในการรับส่งข้อมูลที่แตกต่างกันออกไปApplication Layer เป็นเลเยอร์ที่แอพพลิเคชั่นเรียกใช้โปรโตคอลระดับล่าง ๆ ลงไปเพื่อวัตถุประสงค์ที่แตกต่างกัน เช่น

FTP (File Transfer Protocol) ใช้สำหรับรับส่งแฟ้มข้อมูลระหว่างโฮสต์

SMTP (Simple Mail Transfer Protocol) ใช้รับส่งจดหมายอิเล็กทรอนิคส์ระหว่างโฮสต์

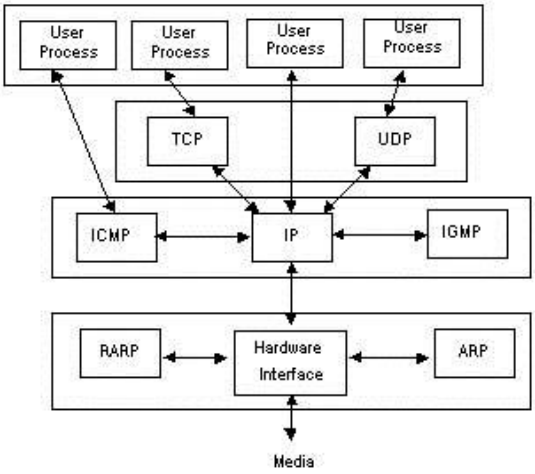
Telnet ใช้สำหรับการควบคุมเครื่องระยะไกล

HTTP (Hypertext Transfer Protocol) เป็นโปรโตคอลที่ใช้รับส่งข้อมูลเว็บเพจระหว่างบราวเซอร์และเว็บเซิร์ฟเวอร์

POP (Post Office Protocol) ใช้สำหรับดาวน์โหลดอีเมล์จากเมล์เซิร์ฟเวอร์มาไว้ที่เครื่องเมล์ไคลเอนต์ (PC) ของผู้ใช้

TCP กับ UDP

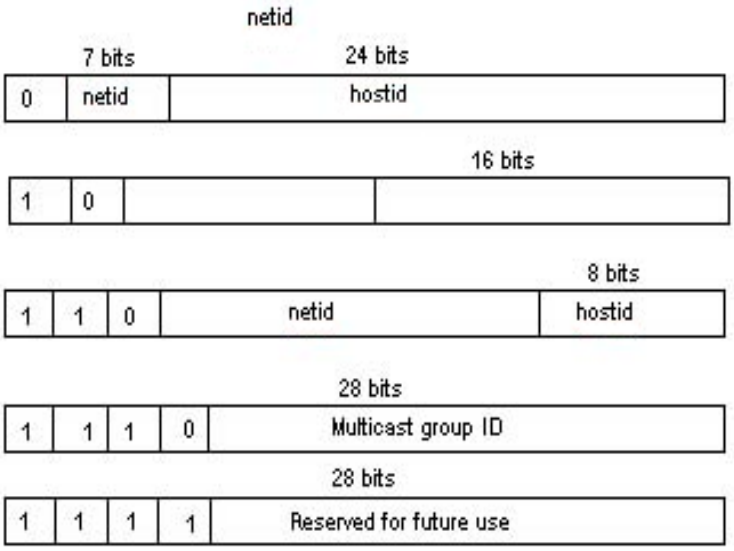
เพื่อเป็นข้อมูลเบื้องต้นสำหรับการทำความเข้าใจชุดโปรโตคอล TCP/IP ก่อนที่จะเจาะลึกรายละเอียดในบทต่อไปนั้นขออธิบายลักษณะของ TCP และ UDP ดังนี้ TCP เป็นโปรโตคอลที่รับประกันการับ - ส่งข้อมูลระหว่างโฮสต์ กล่าวคือ โปรโตคอลมีกลไกในการตรวจสอบและยืนยันว่ามีข้อมูลจากต้นทางจะไปถึงปลายทางเสมอ หากข้อมูลถึงปลายทางก็จะมีสัญญาณตอบรับว่าข้อมูลถึงปลายทางแล้ว หากไม่มีสัญญาณตอบรับก็แสดงว่าข้อมูลไม่ถึงปลายทาง ดั้งนี้แอพพลิเคชั่นที่มีความสำคัญจึงเลือกใช้โปรโตคอลนั้นในการรับส่งข้อมูล กระบวนการยืนยันการรับส่งข้อมูลนี้เองเป็นจุดขายของโปรโตคอล TCP ที่เด่นและไม่มีใครมาแข่งขันได้ เพราะในการทำงานของคอมพิวเตอร์ ท่านผู้อ่านคงทราบแล้วว่าข้อมูลทุกตัวอักษรล้วนมีความสำคัญอย่างยิ่งยวดและอย่างเท่าเทียมกัน การที่ข้อมูลขาดหายหรือผิดไปแม้เพียงตัวอักษรเดียวก็อาจทำให้โปรแกรมทำงานผิดพลาดทั้งได้ทันที ดังนั้นแอพพลิเคชั่นซึ่งทราบความสำคัญในจุดนี้เป็นอย่างดีจึงหันมาเลือกใช้โปโตคอล TCP ในการสื่อสารกันเป็นจำนวนมากประกอบกับการที่โปรโตคอล TCP นั้นอยู่บนเลเยอร์ของ IP ซึ่งสามารถรับส่งข้อมูลไปได้ทุกที่ในโลกก็เปรียบเสมือนเสือติดปีกและเราเองที่ได้เห็นฤทธิ์เดชของเจ้าของเสือติดปีกบนอินเทอร์เน็ตทุกวันนี้เอง ซึ่งโปรโตคอล UDP คือ TCP ที่มีการรับส่งข้อมูลได้อย่างถูกต้องทุกบิต แต่ทุกอย่างในโลกนี้ไม่มีอะไรได้มาฟรี สิ่งที่ TCP ต้องแลกไปกับการรับประกันข้อมูลคือโอเวอร์เฮดที่เพิ่มขึ้นไม่ว่าในแง่ของความยาวของข้อมูลและความซับซ้อนในการที่ต้องการตอบรับทุกครั้งทำให้ประสิทธิภาพลดลงไปโปรโตคอล UDP ได้เข้ามาแก้ไขปัญหานี้คือ สามารถรับส่งข้อมูลระหว่างโฮสต์ได้เช่นกัน แต่ไม่มีการับประกันการรับส่งข้อมูล หมายถึงการส่งข้อมูลทุกครั้งจะไม่มีการตรวจสอบยืนยันกันเอง หากต้องการตรวจสอบก็ให้รับส่งข้อมูลยืนยันข้อมูลอาจจะขอเพียงให้สามารถรับ – ส่งข้อมูลได้อย่างมีประสิทธิภาพก็เพียงพอแล้ว ซึ่ง TCP/IP Layering ในชุดของโปโตคอล TCP/IP ประกอบด้วยโปรโตคอลหลายตัวทำงานร่วมกันในเลเยอร์ต่างๆและมีหน้าที่ แตกต่างกันออกไปแสดงให้เห็นถึงโปรโตคอลในแต่ละเลเยอร์ที่เมื่อรวมกันเป็นชุมของ TCP/IP ซึ่งตัวหลักในภาพก็คือ TCP อยู่ในทราบสปอร์ตเลเยอร์ทำหน้าที่จัดการและควบคุมการรับส่งข้อมูลให้มีเสถียรภาพและเชื่อถือได้โดยปล่อยหน้าที่นี้ให้กับแอพพลิเคชั่นเลเยอร์เป็นผู้ทำหน้าที่นี้แทน UDP อยู่ในสปอร์ตเลเยอร์ ทำหน้าที่จัดการและควบคุมการรับส่งข้อมูลเช่นเดียวกัน แต่ไม่มีกลไกการรับส่งที่มีเสถียรภาพและเชื่อถือได้ โดยปล่อยหน้าที่นี้ให้กับแอพพลิเคชั่นเลเยอร์เป็นผู้ทำหน้าที่นี้แทน



ภาพที่ 2.17 เลเยอร์ของโปรโตคอลต่างๆ ในชุด TCP/IP Suite

IP อยู่ในเน็ตเวิร์กเลเยอร์เป็นโปรโตคอลหลักในการสื่อสารข้อมูลซึ่งกลไกสำคัญที่ทำให้ข้อมูลสามารถเคลื่อนที่ไปยังปลายทางได้ก็คือโปรโตคอล IP นั้นเอง ICMP (Internet Control Message Protocol) อยู่ในเน็ตเวิร์กเลเยอร์ ทำหน้าที่เสริมให้การทำงานของ IP ให้สมบูรณ์ โดยจะเป็นโปรโตคอลที่คอยส่งข่าวสารและแจ้งความผิดพลาดให้แก่ IP แต่ในบางโอกาสแอพพลิเคชั่นเลยเยอร์ก็เรียกใช้ ICMP โดยตรงเพื่อใช้ประโยชน์จากความสามารถของ ICMP ด้วยเช่นกัน IGMP (Internet Group Management Protocol) อยู่ในเน็ตเวิร์กเลเยอร์ ทำหน้าที่ในการส่ง UDP ดาต้าแกรมไปยังกลุ่มของโฮสต์ หรือโฮสต์หลาย ๆ ตัวพร้อมกัน ARP (Address Reservation Protocol) อยู่ในลิงค์เลเยอร์ ทำหน้าที่เปลี่ยนระหว่างแอดเดรสที่ใช้โดย IP ให้เป็นแอดเดรสของ Network Interface RATP (Reverse ARP) อยู่ในลิงค์เลเยอร์เช่นกัน แต่ทำหน้าที่กลับกันกับ ARP คือ เปลี่ยนระหว่างแอดเดรส ของ Network Interface ให้เป็นแอดเดรสที่ใช้โดย IP

Internet Address ทุกอินเทอร์เน็ตทีต่อยู่บนอินทอร์เน็ตจะต้องมีหมายเลขประจำตัวเพื่อใช้ในการสื่อสาร ข้อมูล เรียกว่า Internet Address หรือเรียกย่อๆว่า IPAddress โดยค่า IPAddress นี้จะเป็นหมายเลขจำนวน 32 บิตแต่แทนที่จะกำหนดให้เลขทั้ง 32 บิต นั้นถูกนับต่อเนื่องกันไป ตั้งแต่ 0 - 2 ก็ใช้วิธีการแบ่งหมายเลขดังกล่าวออกเป็นกลุ่มของเลขขนาด 8 บิต จำนวน 4 ชุด และคั่นแต่ละชุดด้วยจุด ตัวอย่างเช่น 192.168.13.201 นอกจากนั้นในIP Address นั้นถูกแบ่งออกเป็น 2 ส่วน คือ ส่วนที่เป็นแอดเดรสของเน็ตเวิร์ก (Network ID)และส่วนที่เป็นแอดเดรสของโฮสต์ (Host ID) ซึ่งข้อมูลในส่วนนี้จะถูกใช้สำหรับค้นหาเส้นทางของ IP ในการที่จะขนส่งข้อมูลจากต้นทางให้ถึงปลายทางอย่างถูกต้องเพื่อเป็นการกำหนดขนาดของเน็ตเวิร์กสำหรับ IP Address ต่าง ๆ ดังนั้นจึงมีการจัด IP Address ในแต่ละช่วงออกเป็นคลาส ต่างๆ กันจาก A ถึง E เพื่อจะได้ทำการจัดสรร IP Address ได้อย่างเหมาะสมกับขนาดของเน็ตเวิร์ก



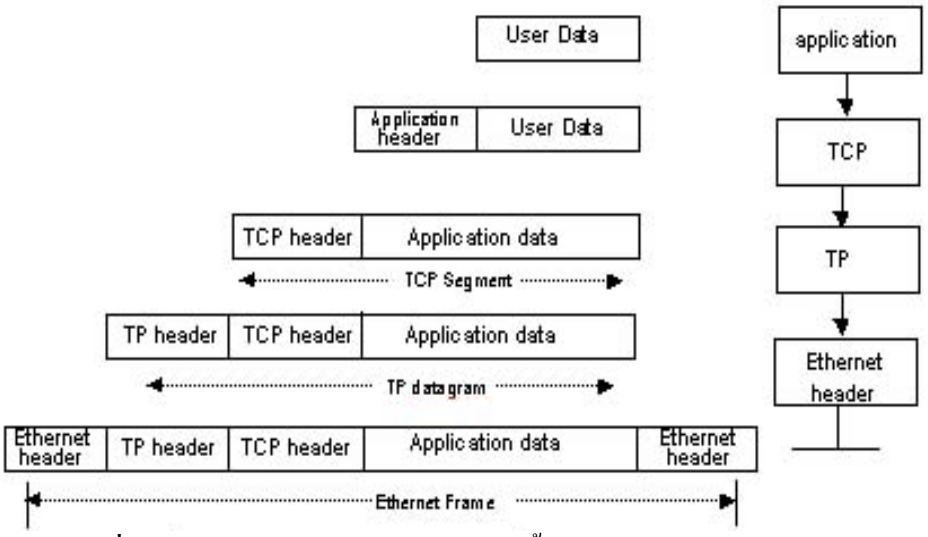
ภาพที่ 2.18 การกำหนดแอดเดรสสำหรับคลาสต่างๆ

Class	Range
A	0.0.0.0 – 127.255.255.255
B	128.0.0.0 – 191.255.255.255
C	192.0.0.0 – 223.255.255.255
D	224.0.0.0.0 – 239.255.255.255
E	240.0.0.0 – 255.255.255.255

ภาพที่ 2.19 แสดงช่วงของ IP Address แต่ละคลาส

การ Encapsulation คือการนำข้อมูลที่ต้องการส่งมาประกอบรวมกับข้อมูลที่เป็นส่วนควบคุมของโปรโตคอลโดยข้อมูลส่วนที่เป็นส่วนควบคุมนั้นจะถูกนำมาไว้ในส่วนหัวข้อของข้อมูลเรียกว่าเฮดเดอร์ (Header)ซึ่งในการรับข้อมูลนั้นผู้รับข้อมูลจะได้รับเฮดเดอร์ก่อนจากนั้นก็นำเฮดเดอร์ไปแปลและทราบว่าข้อมูลที่ตามมานั้นมีลักษณะอย่างไรจะได้จัดการได้อย่างถูกต้อง ภายในเฮดเดอร์ของโปรโตคอลส่วนใหญ่จะประกอบด้วยข้อมูลหลักที่สำคัญของโปรโตคอลที่ทำการ Encapsulate มาคือแอดเดรสต้นทาง, แอดเดรสปลายทาง, ความยาวข้อมูล, รหัสตรวจสอบความผิดพลาดข้อมูลซึ่งสิ่งที่จะต้องเน้นให้เห็นชัดคือ จะมีข้อมูลสำคัญเฉพาะโปรโตคอลที่ทำการ Encapsulation มาเท่านั้น ตัวอย่างเช่น การ Encapsulate ของ Ethernet ก็จะมีการระบุ Ethernet address ลงในเฮดเดอร์เท่านั้นจะไม่มีการบรรจุ IP address ลงมาใน Ethernet Header ด้วยแต่อย่างใดเพราะในเลเยอร์ของ Ethernet จะไม่รู้จัก IP Address หรือรหัสควบคุมใด ๆ ของ IP (จริง ๆ แล้ว

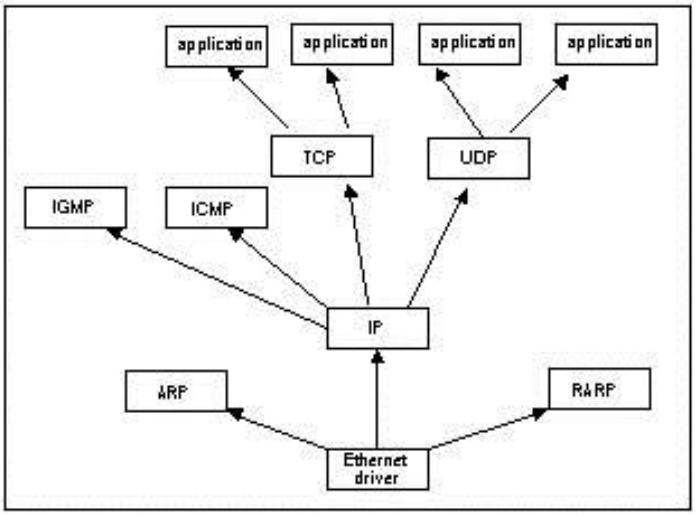
Ethernet ไม่รู้ด้วยซ้ำว่าข้อมูลที่จะส่งนั้นเป็นโปรโตคอลอะไร) ดาต้าแกรมของ IP จะถูกตีค่าว่าเป็นข้อมูลก้อนเดียวกันสำหรับ Ethernet เท่านั้น



ภาพที่ 2.20 การ Encapsulation ข้อมูลผ่านชั้นของโปรโตคอลแต่ละระดับ

ในการับส่งข้อมูลนั้นข้อมูลที่รับส่งกันจริงๆบนเน็ตเวิร์กนั้นจะประกอบด้วย 2 ส่วนคือข้อมูลจริงกับข้อมูลของโปรโตคอลเปรียบเสมือนการส่งจดหมายซึ่งจะต้องประกอบด้วยเนื้อความในจดหมายและซองจดหมายที่เขียนที่อยู่ติดแสตมป์ถ้ามีแต่จดหมายอย่างเดียวไปรษณีย์ก็คงต้องอ่านเองเพราะไม่รู้ว่าเราจะส่งให้ใครการ Encapsulate ก็คือการเอาจดหมายมาใส่ซองนั้นเองโดยซองจะเปรียบเสมือนข้อมูลที่ใช้ในการรับส่งข้อมูลของโปรโตคอลนั้น 1 โปรโตคอลก็กจะใส่ 1 ซอง ถ้าข้อมูลต้องส่งผ่านหลายเลเยอร์ จำนวนซองก็จะถูกใส่เพิ่มหลายชั้นตามลำดับการ Encapsulate นั้นเองดังนั้นถ้าเราจะส่งข้อมูลผ่านโปรโตคอล TCP ข้อมูลเราก็จะถูกใส่ซองตามลำดับดังนี้ ซอง TCP, ซอง IP, ซอง Ethernet และฝ่ายรับข้อมูลก็จะต้องแกะซองออกตามลำดับ โดยต้องแกะซอง Ethernet ก่อนแล้วจะเจอซอง IP แกะซอง IP จะเจอซอง TC Pแกะซอง TCP ก็จะเจอข้อมูลที่ต้องการการ Encapsulateในแต่ละระดับก็จะมีการเรียกข้อมูลที่อยู่ในซองแตกต่างกันออกไป ข้อมูลทำการ Encapsulate เรียบร้อย แล้วจาก TCP ส่งไปยัง IP เรียกว่า TCP Segment ในระดับ IP ก็จะถือว่า TCP Segment เป็นข้อมูลทั้งชุด เมื่อไปรวมกับ IP Header ส่งไปยังเลเยอร์ Data Link จะเรียกว่า IP Datagram ในระดับ Data link เมื่อส่งลงไปจะนำ IP Datagram มาใส่ซองขนาดของข้อมูลทั้งหมดเราจะเรียกว่า Ethernet Frame จากรูปจะเห็นว่าบางครั้งข้อมูลเรามีอยู่เพียงเล็กน้อยแต่กว่าที่เราจะส่งข้อมูลไปถึงปลายทางได้จะมีข้อมูลของเฮดเดอร์ของโปรโตคอลเกาะติดไปด้วยเสมอ เช่นเดียวกับที่บางครั้งเราได้รับของขวัญที่กล่องใหญ่ห่อหลายชั้น แต่พอเปิดไปข้างในอาจจะมีเพียง

ช็อคโกแลตแท่งเดียว เป็นต้น ซึ่งก็เป็นได้บางครั้งอาจจะดูสิ้นเปลืองแต่การที่ต้องใส่ของหลายชั้นแล้วส่งที่หมายดีกว่าประหยัดซองแต่จดหมายไปถึงสำหรับในการสื่อสารข้อมูลไม่มีซองของจริงๆ ให้สิ้นเปลืองแต่ทรัพยากรที่เราสิ้นเปลืองไปก็คือแบนด์วิดธ์ที่เราอาจใช้งานได้ไม่เต็มประสิทธิภาพเท่าที่ควรเนื่องจากทุกแพ็กเก็ตของข้อมูลจะต้องเสียส่วนหนึ่งไปเป็นเฮดเดอร์เสมออาจสังเกตได้ง่ายๆ ว่าในขณะที่โมเด็มต่อกับอินเทอร์เน็ตด้วยอัตราการรับส่งข้อมูล 56 Kbps แต่ความเร็วที่เราดาวน์โหลดไฟล์ถึงได้สูงสุดต่ำกว่านั้นมากเช่น 30 Kbps สาเหตุก็เนื่องมาจากการที่มีเฮดเดอร์โปรโตคอลรวมอยู่กับข้อมูลจริงอยู่ด้วย ทำให้ต้องเสียแบนด์วิดธ์ไปส่วนหนึ่ง



ภาพที่ 2.21 Dumultiplexing

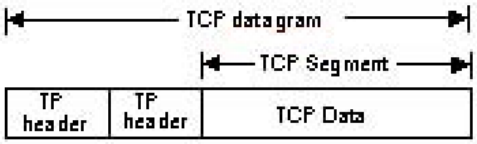
ในหัวข้อที่ผ่านมาได้กล่าวถึงการ Encapsulation ไปแล้ว Demultiplexing คือกระบวนการย้อนหลังของการ Encapsulation นั่นเองหากกระบวนการ Encapsulation คือการนำข้อมูลมาใส่ของที่ละชั้นตามเลเยอร์ที่ส่งไปการ Demultiplexing ก็คือการรับซองข้อมูลที่ปิดผนึกใส่ซองมาอย่างมิดชิด เพื่อทำการแกะออกที่ละขั้นตามเลเยอร์จนถึงเลเยอร์สุดคือ แอพพลิเคชั่นเลเยอร์ จึงได้ข้อมูลเนื้อความจริง ๆที่ต้องการสื่อสารกันในการ Demultiplexing นั้นแต่ละเลเยอร์จะนำข้อมูลมารวมกันให้ครบตามขนาดที่ต้องการเช่น 1 เฟรมในระดับ Ethernet 1 Datagram ในระดับ IP และ 1 Segment ในระดับ TCP และในแต่ละเลเยอร์ก็จะทำการถอดเฮดเดอร์ซึ่งเปรียบเสมือนซองออกแล้วส่งขึ้นไปบนเลเยอร์ที่สูงกว่า ซึ่งในที่สุดเลเยอร์สุดท้ายก็จะได้รับเฉพาะข้อมูลเท่านั้น และเฮดเดอร์ถูกถอดออกไปหมดการ Demultiplex และ Encapsulate เป็นสิ่งคู่กันและสอดคล้องกัน อุปกรณ์ที่จะสื่อสารบนเน็ตเวิร์กได้จะต้องมีทั้งส่วนที่ทำหน้าที่ทั้งสอง โดยการ Demultiplexing ใช้ในตอนที่รับข้อมูลจากเน็ตเวิร์ก และการ Encapsulateใช้ในตอบที่จะทำการส่งข้อมูลอยู่ในทุกเลเยอร์ของโปรโตคอล

Port Number จากเลเยอร์ที่แสดงในภาพจะเห็นว่าเลเยอร์บนสุดของ TCP/IP คือ แอพพลิเคชั่นเลเยอร์ สิ่งหนึ่งที่เราสามารถสังเกตเห็นได้ว่าข้อมูลทั้งหมดทุกๆ เซกเมนต์จะต้องผ่านเลเยอร์นี้ก็เพราะแอพพลิเคชั่นเลเยอร์จะครอบคลุมทั้งหมด ดังนั้นจะเกิดอะไรขึ้นถ้ามีหลายแอพพลิเคชั่นต่างก็ต้องการรับส่งข้อมูลผ่าน TCP/IP แต่ละแอพพลิเคชั่นจะสามารถแยกแยะได้อย่างไรในความเป็นจริงที่ใช้งานปัจจุบันเองก็มีอยู่มากที่มีแอพพลิเคชั่นมากกว่า 1 แอพพลิเคชั่นที่ทำงานอยู่ภายในเครื่องเดียวกันเช่นในเซิร์ฟเวอร์เครื่องเดียวอาจจะเป็นทั้ง FTP server .Web server และ Mail server นอกจากนี้อาจจะมีการบริการอื่น ๆ บน TCP/IP ที่ซ่อนอยู่โดยที่เราอาจไม่ทราบ เช่น NetBIOS เป็นต้นพอร์ต (port) จะเป็นปัญหาของคำถามข้างต้นในโปรโตคอล TCP/IP ได้ถูกออกแบบให้ มีหมายเลขพอร์ตอยู่ในเฮดเดอร์เพื่อระบุว่าข้อมูลแซกเมนต์นี้เป็นของแอพพลิเคชั่นอะไรในโฮสต์นั้น แอพพลิเคชั่นแต่ละตัวที่ให้บริการอยู่ในเครื่องต่างจะมีหมายเลขพอร์ตประจำตัวเพื่อจะสามารถเลือกนำข้อมูลมาใช้ว่าเป็นของแอพพลิเคชั่นตนเองหรือไม่หมายเลขพอร์ตที่เรารู้จักกันดีและใช้เป็นมาตรฐาน ได้แก่พอร์ต 20, 21 เป็นของ FTP, พอร์ต 23Telnet. พอร์ต 25SMTP, พอร์ต 30 HTTP เป็นต้นสำหรับรายละเอียดว่าพอร์ตใดสำหรับแอพพลิเคชั่นเปิดดูได้ในภาคผนวกท้ายเล่ม โดยทั่วไปหมายเลขพอร์ตจะมีความสำคัญกับฝั่งของเซิร์ฟเวอร์เท่านั้น เนื่องจากแอพพลิเคชั่นฝั่งเซิร์ฟเวอร์จะต้องคอยรอรับการรีเควสท์ หรือขอรับบริการจากไคลแอนต์ที่พอร์ตเดิมเสมอ ส่วนในฝั่งไคลแอนต์เองหมายเลขพอร์ตไม่จำเป็นต้องเป็นหมายเลขที่ตายตัว และคงที่เพราะหมายเลขพอร์ตจะเป็นการสุ่มเลขขึ้นมาใช้ชั่วคราว และจะมีการเรียกใช้พอร์ตใหม่ทุกครั้งที่มีการรับส่งข้อมูลเซสชั่น ใหม่

TCP Header

ใน TCP Header จะเริ่มต้นระบุที่หมายเลขพอร์ตต้นทางและหมายเลขพอร์ตปลายทาง แต่อันที่จริงแล้วข้อมูลอีกส่วนหนึ่งที่ใช้ในการสื่อสารคือ IP Address ของต้นทางและปลายทางก็ต้องระบุเช่นกัน แต่ได้ถูก Encapsulateไว้ในเลเยอร์ของ IP และค่าของ IP Address ทั้งคู่จะอยู่ใน TCP Header ส่วนคู่ของ IP Address และหมายเลขพอร์ตนั้นจะเรียกว่า “ซ็อคเก็ต” ซึ่งในการสื่อสารแต่ละครั้งจะต้องมีทั้งซ็อคเก็ตของต้นทางและปลายทางข้อมูลจึงจะถูกส่งรับไปถูกที่และถูกแอพพลิเคชั่น รายละเอียดหน้าที่ของแต่ละฟิลด์ใน TCP Header มีดังนี้ SourcePortNumber หมายถึง พอร์ตที่โฮสต์ต้นทางใช้ในการสื่อสารของเซสชั่นนี้และ TCP จะใช้พอร์ตนี้ไปตลอดตราบใดที่การสื่อสารในเซสชั่นนี้ยังไม่ยุติลง โดยทั่วไปนี้จะเรียกว่า “ไคลเอนต์พอร์ต” คือ พอร์ตที่ไคลเอนต์เปิดขึ้นมาเพื่อรอการตอบรับจากเซิฟร์เวอร์ไคลเอนต์พอร์ตจะมีหมายเลขไม่แน่นอนและเปลี่ยนไปทุกครั้งที่มีการเชื่อมต่อใหม่เป็นพอร์ตที่ถูกเปิดไว้ในระยะเวลาสั้นๆค่าที่เป็นไปได้ของพอร์ตนี้ขึ้นอยู่กับการ

จัดสรรค์ของระบบปฏิบัติการในการกำหนดขอบเขตของพอร์ตเหล่านี้ส่วนใหญ่จะมีค่าอยู่ในช่วง 1024 - 5000 Destination Port Number หมายถึง หมายเลขพอร์ตบนโฮสต์ปลายทางที่โฮสต์ต้นทางต้องการติดต่อด้วยโดยนัยแล้วหมายถึงแอพพลิเคชั่นที่ให้บริการอยู่บนพอร์ตนั้นที่โฮสต์ปลายทางนั่นเองพอร์ตนี้จะเรียกอีกอย่างหนึ่งว่า “เซิร์ฟเวอร์พอร์ต” หมายเลขพอร์ตที่เปิดไว้จะขึ้นอยู่กับแอพพลิเคชั่นที่ให้บริการ โดยทั่วไปแอพพลิเคชั่นแต่ละประเภทจะมีหมายเลขพอร์ตเป็นมาตรฐานสำหรับให้ไคลเอนต์ได้เรียกให้บริการ Sequence Number เป็นฟิลด์ที่ระบุหมายเลขลำดับที่ใช้อ้างอิงในการสื่อสารข้อมูลแต่ละครั้งเพื่อให้ทั้ง 2 ฝ่ายจะได้รับทราบตรงกันว่าเป็นข้อมูลของชุดใดการนำไปใช้งานจะได้ไม่ปะปนกันและมีลำดับที่ถูกต้องเนื่องจากการสื่อสารข้อมูลผ่าน TCP นั้นจังหวะและลำดับเป็นส่วนสำคัญของโปรโตคอลไม่ยิ่งหย่อนไปกว่าข้อมูลใน TCP Header รวมไปถึงการที่ข้อมูลในแต่ละ TCP Segment อาจจะถูกแบ่งออกและส่งไปในลำดับที่ไม่เรียงกันหากไม่มีจุดอ้างอิงของข้อมูล ก็จะไม่สามารถอ่านข้อมูลกลับใหม่ได้อย่างสมบูรณ์และถูกต้อง การส่งข้อมูลและการตอบรับจะใช้ฟิลด์นี้เป็นตัวยืนระหว่างกันเสมอ



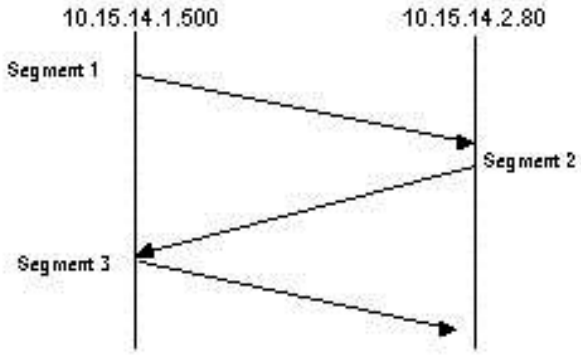
ภาพที่ 2.22 Encapsulation ของข้อมูล TCP ใน IP Datagram

16 – bit source port number	16 – bit source port number
32 – bit sequence number	
32 – bit acknowledge number	
4-bit header length / reserved (6 bits) / URG ACK PSH RST SYN FIN	
16 – bit TCP Checksum	16 – bit urgent pointer
Option (ถ้ามี)	
Data (ถ้ามี)	

ภาพที่ 2.23 TCP Header

Acknowledge Number ทำหน้าที่เช่นเดียวกับ Sequence Number ต่างกันตรงที่เป็น Sequence Number ซึ่งใช้ในการตอบรับกล่าวคือเนื่องจาก Sequence Number ที่ใช้ในการอ้างอิงนั้นผู้ที่เริ่มส่งข้อมูลจะเป็นผู้กำหนดเลขขึ้นมาและส่งไปพร้อมกับการสร้างการเชื่อมต่อครั้งใหม่แต่

สำหรับฝ่ายที่ถูกติดต่อก็จำเป็นต้องกำหนดหมายเลขสำหรับใช้อ้างอิงในการตอบรับเช่นกัน ค่าที่อยู่ใน Acknowledge Numberก็คือหมายเลขที่ใช้อ้างอิงในการตอบรับนี้ ปกติค่าทั้งใน Sequence Number จะต้องพิจารณาประกอบกับ Flag จึงจะสามารถแปลความหมายของ TCP Segment ได้อย่างสมบูรณ์ Header Length โดยปกติความยาวของ TCP Header จะเท่ากับ 20ไบต์แต่ถ้าหากมีการใช้ค่า Option อาจจะทำให้ขนาดของเฮดเดอร์ยาวขึ้นตามข้อมูลที่ต้องเพิ่มมาจาก Option นั้น แต่ทั้งหมดแล้วจะไม่เกิน 60 ไบต์ Flag เป็นข้อมูลในระดับบิตที่ใช้เป็นตัวบอกคุณสมบัติของ TCP Segment ที่กำลังส่งอยู่นั้น และใช้เป็นตัวควบคุมจังหวะการรับส่งข้อมูลด้วย ซึ่ง Flag ทั้งหมดมีอยู่ 6 บิต แต่ละบิตมีชื่อและมีความหมายดังนี้ URG แสดงว่าข้อมูลในฟิลด์ Urgent Pointer นั้นนำมาใช้งานได้ ACK แสดงว่าข้อมูลในฟิลด์ Acknowledge Number นำมาใช้งานได้ DSH เพื่อแจ้งให้ผู้รับข้อมูลทราบว่าควรจะส่งข้อมูล Segment นี้ไปยังแอพพลิเคชั่นโดยเร็ว RST ยกเลิกการติดต่อ SYN ใช้ในการเริ่มต้นขอติดต่อกับปลายทาง FIN ใช้ส่งเพื่อแจ้งให้ปลายทางทราบว่ายุติการติดต่อ Window Size เป็นขนาดของการรับ - ส่งข้อมูลในแต่ละครั้งที่ทางฝ่ายผู้รับจะสามารถรับได้ เนื่องจากในการรับข้อมูลนั้น ทางผู้รับจะต้องจัดเตรียมหน่วยความจำในการพักข้อมูลที่มาจาก TCP และทำการ Demultiplex ออกมาหากไม่มีการตกลงถึงขนาดที่ทางฝ่ายรับสามารถรับได้ ก็จะทำให้การสื่อสารข้อมูลไม่สมดุล และฝ่ายรับอาจจะประมวลผลไม่ทันซึ่งส่งผลให้ต้องส่งข้อมูลหลายครั้ง Checksum ฟิลด์ที่ใช้ในการตรวจสอบ ความถูกต้องของข้อมูลใน TCP เซกเมนต์ Urgent Pointer ใช้ระบุหมายเลข Sequence Number ของ TCP เซกเมนต์ล่าสุดที่อยู่ในโหมด Urgent Option ข้อมูลเพิ่มเติมซึ่งจะอยู่ใน TCP Header เมื่อมีการตั้งค่า option บางอย่างที่ต้องการข้อมูลเพิ่มเติมซึ่งไม่มีใน TCP Header เช่น MSS, Strict Route Connection Establishment และ Termination



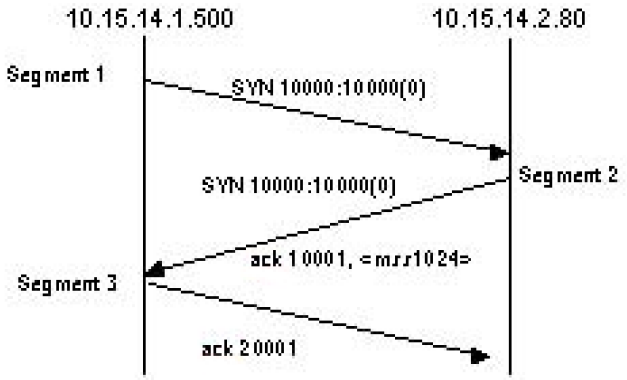
ภาพที่ 2.24 Connection Establishment

ก่อนที่ TCP จะสามารถรับส่งข้อมูลได้จะต้องมีการสถาปนา (Establishment) หรือการสร้างให้มี Connection เกิดขึ้นก่อนเปรียบเสมือนการต่อสายของทั้ง 2 ฝั่งให้เชื่อมถึงกันซึ่งโปรโตคอล TCP ได้กำหนดขั้นตอนในการเริ่มต้นสร้าง Connection ไว้ดังนี้

1. เครื่องไคเอนต์จะทำการส่งเซกเมนต์โดยเปิด SYNFlag ระบุหมายเลขที่ต้องการติดต่อบนเซิร์ฟเวิร์กและระบุหมายเลขลำดับของข้อมูล (ISN - Initial Sequence Number)

2. เครื่องเซิร์ฟเวอร์เมื่อได้รับข้อมูลเซกเมนต์จากข้อ 1 ก็จะตอบกลับด้วยการเพิ่มค่า ISN ที่ไดรับอีกขึ้นอีก 1 พร้อมทั้งระบุหมายเลขลำดับของตนเอง และเปิด SYN กับ ACK Flag

3. ไครเอนต์เมื่อได้รับการตอบกลับจากเซิร์ฟเวอร์ตามข้อ 2 ก็จะทำการตอบรับกลับไป โดยการเพิ่มค่า ISN ของเซิร์ฟเวอร์ขึ้นอีก 1 และเปิด ACKFlag เมื่อผ่านการสร้าง Connection ทั้ง 3 ขั้นตอนแล้วตอนนี้ทั้งไคลเอนต์และเซิร์ฟเวอร์เปรียบเสมือนมีการเชื่อมต่อถึงกันแล้วสามารถส่งรับข้อมูลกันได้ตลอดจนกว่าจะมีการยุติ Connection นั้นเสีย ขั้นตอนทั้ง 3 เรียกว่า “Three - Ways Handshakes”

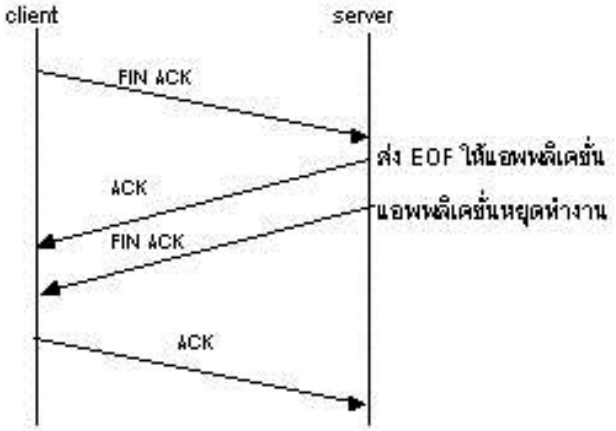


ภาพที่ 2.25 Connection Three - Ways Handshakes

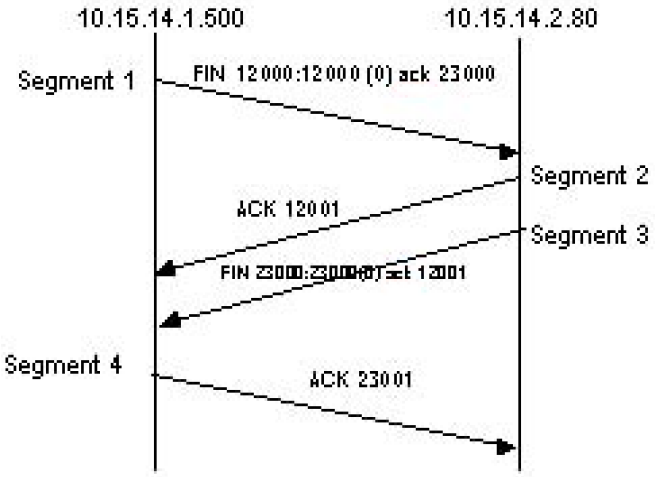
จากภาพที่ 2.25 จะแสดงโฮสต์ในลักษณะ Hostname port เป็นการแสดงให้เห็นขั้นตอนการสร้าง Connection ระหว่างโฮสต์ 10.15.14.1 จากพอร์ตหมายเลข 500 ทำหน้าที่เป็นไคลเอนต์ส่งคำสั่งไปยังโฮสต์ 10.15.14.2 พอร์ต 80 ซึ่งเป็นเว็บเซิร์ฟเวอร์ โดยจากข้อมูลอธิบายได้ดังนี้ Segment 1 โฮสต์ 10.15.14.1 ใช้พอร์ตหมายเลข500เป็นไคลเอนต์พอร์ตได้ส่งสัญญาณเพื่อขอเริ่มการเชื่อมต่อโดยการเซ็ต TCP Flag SYN ไปยัง 10.15.14.2 พอร์ตหมายเลข 80 มีหมายเลข Sequence เท่ากับ 10000 ซึ่ง Connection Establishment Segment2 โฮสต์ 10.15.14.2 ได้ตอบรับการ SYN ของ 10.15.14.1 โดยการส่งสัญญาณตอบกลับไปยังโฮสต์ 10.15.14.1 หมายเลขพอร์ต 500 โดยเซ็ต TCP Flag SYN และ TCP Header และใช้หมายเลข Sequence เท่ากับ 20000 และหมายเลข Acknowledge

เท่ากับ 1001 แสดงว่าโฮสต์ 10.15.14.2 ตอบรับคำขอเชื่อมต่อของ โฮสต์ 10.15.14.1 แล้วโดยจะใช้ช็อคเก็ตนี้ในการสื่อสารต่อไป Segment3 โฮสต์ 10.15.14.1 เมื่อได้รับการตอบรับจากโฮสต์ 10.15.14.2 ก็ทำการยืนยันให้ทราบว่าได้รับการตอบรับแล้วโดยการส่งสัญญาณตอบไปยังพอร์ต 80 โดยเซ็ต TCP flag ACK และใช้หมายเลข Acknowledge 20001 (200011 = 20000) หลังจากกระบวนการผ่านไปทั้ง 3 เซกเมนต์แล้วแสดงว่า Connection ได้ถูกจัดตั้งเรียบร้อยแล้วหลังจากนั้นก็สามารถเริ่มสื่อสารได้โดยการสื่อสารจะอยู่ในช็อคเก็ตของ 10.15.14.1.500 ถึง 10.15.14.2.80 สำหรับจากตัวอย่างนี้ โฮสต์ 10.15.14.1 เป็นผู้ริเริ่มการติดต่อจะต้องเปิดพอร์ตภายในโฮสต์ตนเองไว้ก่อน เรียกว่า “Active Open” สำหรับโฮสต์ 10.15.14.2 เป็นผู้ตอบสนองต่อการติดต่อนั้นเมื่อยินยอมติดต่อด้วยก็จะต้องเปิดพอร์ตเช่นกัน เรียกว่า “Passive Open” Connection Termination หลังจากการรับส่งข้อมูลได้ยุติลง จะต้องทำขั้นตอนยุติการรับ - ส่งข้อมูล เปรียบเสมือนการถอดสายเชื่อมต่อที่อยู่ระหว่างไคลเอนต์กับเซิร์ฟเวอร์ออกไปการสิ้นสุดการรับส่งข้อมูลโดยสมบูรณ์มีอยู่ 4 ขั้นตอน คือ

1. ไคลเอนต์ทำการส่ง ISN พร้อมกับ FIN ACK Flag ไปยังเซิร์ฟเวอร์
2. เซิร์ฟเวอร์ทำการตอบรับ ISN และบวกค่า ISN อีก 1 พร้อมกับ ACK Flag
3. เซิร์ฟเวอร์ทำการส่ง ISN พร้อมกับ FIN ACK Flag ไปยังไคลเอนต์
4. ไคลเอนต์ตอบรับการยุติการสื่อสารด้วย ISN + 1 พร้อมกับ ACK Flag



ภาพที่ 2.26 Connection Termination 1



ภาพที่ 2.27 Connection Termination 2

จากภาพที่ 2.26 และภาพที่ 2.27 แสดงให้เห็นการยุติการเชื่อมต่อสามารถอธิบายการทำงานในแต่ละเซกเมนต์ของการยุติการติดต่อได้ดังนี้ Segment 1 พอร์ตหมายเลข 500 บนโฮสต์ 10.15.14.1 ต้องการยุติการส่งข้อมูลให้กับพอร์ตหมายเลข 80 ของโฮสต์ 10.15.14.2 จึงทำากรส่งสัญญาณโดยการเซ็ต TCP flag FIN ของ TCP เซกเมนต์ โดยหมายเลข Sequence เท่ากับ 12000 ไปยังโฮสต์ 10.15.14.2 พอร์ต หมายเลข 80 Segment 2โฮสต์ 10.15.14.2 ได้รับสัญญาณยุติการติดต่อ (FIN) ก็ทำการตอบรับโดยการส่ง ACK กลับไปพร้อมหมายเลข Acknowledge เท่ากับ 12001 เพื่อเป็นการรับทราบการยุติการส่งข้อมูลของโฮสต์ 10.15.14.1 Segment 3 โฮสต์ 10.15.14.2 ได้ส่งสัญญาณยุติการส่งข้อมูลไปยังโฮสต์ 10.15.14.1 เช่นกัน โดยการส่ง FIN ด้วยหมายเลข Sequence เท่ากับ 23000 Segment 4 โฮสต์ 10.15.14.1 ได้รับสัญญาณยุติการส่งของโฮสต์ 10.15.14.2 ในเซกเมนต์ 3 ก็ทำการตอบรับกลับไปด้วย ACK หมายเลข Acknowledge เท่ากับ 23001 ครบทั้ง 4 เซกเมนต์เป็นอันหมดสิ้นกระบวนการยุติการติดต่อระหว่างโฮสต์ 10.15.14.2 และโฮสต์ 10.15.14.1 เรียบร้อยหากต้องการจะสื่อสารกันใหม่ก็ต้องส่งสัญญาณเพื่อสร้างการเชื่อมต่อกันใหม่การยุติการติดต่อสามารถกระทำได้ทั้งฝั่งไคลเอนต์หรือเซิร์ฟเวอร์ และสามารถยุติเวลาใดของการติดต่อก็ได้ TCP Half - Close การยุติการเชื่อมต่อโดยที่โฮสต์ส่ง FIN ACK ออกไป มิได้หมายความว่าการสื่อสารจะยุติลงในทันที เป็นเพียงการแจ้งให้อีกฝ่ายหนึ่งทราบว่าจะไม่มีข้อมูลส่งอีกต่อไป แต่โฮสต์จะยังคงสามารถรับข้อมูลเข้ามาได้อีกต่อไป จนกว่าโฮสต์อีกฝั่งหนึ่งจะส่ง FIN ACK เพื่อขอยุติการส่งข้อมูลเช่นกัน เมื่อไม่มีฝ่ายใดประสงค์จะส่งข้อมูลกันนั่นก็คือการเชื่อมต่อได้ยุติลงอย่างสมบูรณ์ ในระหว่างที่โฮสต์ส่ง FIN ACK ออกไปและยังไม่ได้รับ FIN ACK กลับมานั้นเรียกว่า Half - Close คือการปิดเพียงด้านส่งข้อมูลเพียงด้านเดียวแต่ยังคงเปิดด้านรับไว้รับข้อมูลซึ่งคุณสมบัตินี้

แอพพลิเคชั่นจะสามารถนำไปใช้ให้เป็นประโยชน์ได้ขึ้นอยู่กับลักษณะของการใช้งานเช่นการสื่อสารข้อมูลบางประเภทโฮสต์ที่ทำหน้าที่รับข้อมูลอาจไม่มีความจำเป็นต้องโต้ตอบกับโฮสต์ที่ส่งข้อมูลในระดับแอพพลิเคชั่น ดังนั้น จึงอาจจะสามารถทำงานได้อย่างต่อเนื่องแม้ว่า TCP จะอยู่ในสถานะ Half - Close ก็ตาม