



ระบบถ่ายโอนข้อมูลแบบกึ่งปิด



วิทยานิพนธ์เสนอบัณฑิตวิทยาลัย มหาวิทยาลัยนเรศวร
เพื่อเป็นส่วนหนึ่งของการศึกษา หลักสูตรวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
ปีการศึกษา 2566
ลิขสิทธิ์เป็นของมหาวิทยาลัยนเรศวร

ระบบถ่ายโอนข้อมูลแบบกึ่งปิด



วิทยานิพนธ์เสนอบัณฑิตวิทยาลัย มหาวิทยาลัยนเรศวร
เพื่อเป็นส่วนหนึ่งของการศึกษา หลักสูตรวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
ปีการศึกษา 2566
ลิขสิทธิ์เป็นของมหาวิทยาลัยนเรศวร

วิทยานิพนธ์ เรื่อง "ระบบถ่ายโอนข้อมูลแบบกึ่งปิด"

ของ อนุรักษ์ นันทา

ได้รับการพิจารณาให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการสอบวิทยานิพนธ์
(รองศาสตราจารย์ ดร.สาคร เมฆรักษาวิช)

..... ประธานที่ปรึกษาวิทยานิพนธ์
(ผู้ช่วยศาสตราจารย์ ดร.ธนธร พอค้า)

..... กรรมการผู้ทรงคุณวุฒิภายใน
(ผู้ช่วยศาสตราจารย์ ดร.วันสุรีย์ มาศกรั่ม)

..... กรรมการผู้ทรงคุณวุฒิภายใน
(ผู้ช่วยศาสตราจารย์ ดร.เกียรียงศักดิ์ เตมีย์)

อนุมัติ

.....
(รองศาสตราจารย์ ดร.กรองกาญจน์ ชูทิพย์)

คณบดีบัณฑิตวิทยาลัย

ชื่อเรื่อง	ระบบถ่ายโอนข้อมูลแบบกึ่งปิด
ผู้วิจัย	อนรรักษ์ นันตา
ประธานที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.ธนธร พ่อคำ
ประเภทสารนิพนธ์	วิทยานิพนธ์ วท.ม. วิทยาการคอมพิวเตอร์, มหาวิทยาลัยนเรศวร, 2566
คำสำคัญ	ไวรัสคอมพิวเตอร์, ระบบไซเบอร์-กายภาพ, การเข้ารหัส AES, ไฟร์ วอลล์, มัลแวร์

บทคัดย่อ

งานวิจัยนี้ได้พัฒนาระบบถ่ายโอนข้อมูลแบบกึ่งปิดใช้ในการถ่ายโอนข้อมูลจากเครื่องคอมพิวเตอร์ที่ไม่มีการเชื่อมต่อกับเครื่องอื่นๆ ภายในองค์กร เป็นการลดความเสี่ยงจากการนำสื่อจัดเก็บข้อมูลจากภายนอกเข้ามาเชื่อมต่อ และลดปริมาณการใช้สื่อจัดเก็บข้อมูลที่เป็นวัสดุสิ้นเปลือง เช่น CD หรือ DVD โดยระบบที่พัฒนาขึ้นจะเชื่อมต่อกับเครื่องคอมพิวเตอร์ที่ไม่มีการเชื่อมต่อกับเครื่องอื่นๆ โดยตรงแต่จะเชื่อมต่อในลักษณะแบบกึ่งปิด เป็นการหลีกเลี่ยงความเสี่ยงที่อาจเกิดขึ้นได้จากการเชื่อมต่อเข้ากับเครือข่ายแบบทั่วไป โดยระบบดังกล่าวจะอาศัยการควบคุมการเปิดหรือปิดของอินเทอร์เน็ตเฟสของตัวกลางสลับกัน โดยอินเทอร์เน็ตเฟสหนึ่งจะเชื่อมต่อกับเครือข่ายแบบปิด ที่มีคอมพิวเตอร์ที่ไม่มีการเชื่อมต่อกับเครื่องอื่นๆ เชื่อมต่อกับคอมพิวเตอร์สำหรับจัดเก็บไฟล์ข้อมูลเพื่อรอทำการโอนถ่ายข้อมูล และอีกหนึ่งอินเทอร์เน็ตเฟสจะเชื่อมต่อกับเครือข่ายท้องถิ่นส่งข้อมูลไปยังเครื่องคอมพิวเตอร์แม่ข่ายสำหรับให้บริการข้อมูล นอกจากนี้การทดสอบหาความสัมพันธ์ขนาดไฟล์กับเวลาในการโอนถ่ายข้อมูลให้สอดคล้องกับการตั้งเวลาเปิดปิดของอินเทอร์เน็ตเฟสทำเวลาเฉลี่ยร่วมที่ดีที่สุดที่ 62.59 วินาที โดยมีขนาดไฟล์อยู่ระหว่าง 500-560 MB ดังนั้นระยะเวลาเปิดและปิดอินเทอร์เน็ตเฟสที่ 60 วินาที และแบ่งขนาดไฟล์ที่ 550MB จึงถูกนำมาใช้เป็นพารามิเตอร์ในการพัฒนาระบบ ระบบที่ได้จะรองรับการโอนถ่ายข้อมูลที่มีประสิทธิภาพและข้อมูลที่ได้รับจากระบบมีความถูกต้อง ได้ถูกนำเสนอในงานวิจัยนี้

Title	SEMI-CLOSED DATA TRANSFER SYSTEM
Author	Anuruk Nunta
Advisor	Assistant Professor Thanathorn Phoka, Ph.D.
Academic Paper	M.S. Thesis in Computer Science, Naresuan University, 2023
Keywords	Virus computer, Cyber-Physical Systems, Advanced Encryption Standard, Firewall, Malware

ABSTRACT

The goal of this research is to develop a semi-closed data transfer system to move data from a network-isolated computer to other machines within an organization. This security measure helps mitigate the risk from USB malware attacks and reduce the consumption of optical storage media, e.g. CDs or DVDs. The proposed system connects to the isolated computer directly without any direct links to other machines due to its open-and-close control interface. The system, which acts as an air gap, allows a file to transfer from the isolated computer into a file storage while it is not connected to other machines. Once this transfer is completed, the system disconnected from the isolated computer and reconnected to other machines. The experimental results show the system feasibility and performance. The correlation between file sizes and open-and-close interface durations were analyzed to optimize for the file-size segmentation. The best average time was 62.59 seconds with the file sizes ranging 500-560 MB. Thus, the open-and-close interface duration of 60 seconds with the file size of 550MB were used as the parameters to develop a system that supports efficient data transfer.

ประกาศคุณูปการ

งานวิจัยเรื่องระบบถ่ายโอนข้อมูลแบบกึ่งปิดประสบความสำเร็จได้ด้วยความกรุณาของศูนย์ปฏิบัติการเครื่องมือวิทยาศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยนเรศวร ที่กรุณาอำนวยความสะดวกช่วยเหลือในการทดสอบและใช้งานระบบที่พัฒนา ขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์.ดร.ธนธร พอค้า ที่ได้กรุณาให้คำปรึกษา แนะนำทางด้านวิชาการ ตลอดจนชี้แนะแนวทางแก้ไขปัญหาอุปสรรค ข้อบกพร่องต่างๆในระหว่างดำเนินการวิจัย ขอกราบขอบพระคุณ คุณธนวัน ม่วงดี หัวหน้างานนโยบาย และแผน ที่ได้ให้คำปรึกษาด้านการวิเคราะห์ทางสถิติ คุณค่าและประโยชน์ของงานวิจัยฉบับนี้ ผู้วิจัยขอขอบเป็นกตัญญูตเวทีแก่ บิดา มารดา ครู อาจารย์ ทุกคนที่มีส่วนในการวางรากฐานการศึกษาให้แก่ผู้วิจัย ตลอดจนผู้ที่มีส่วนเกี่ยวข้องทุกท่าน ที่กรุณาให้ความช่วยเหลือและเป็นกำลังใจแก่ผู้วิจัย จนกระทั่งทำให้งานวิจัยฉบับนี้สำเร็จลุล่วงไปด้วยดี

อนุรักษ์ นันทา



สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ค
บทคัดย่อภาษาอังกฤษ.....	ง
ประกาศคุณูปการ.....	จ
สารบัญ.....	ฉ
สารบัญตาราง.....	ญ
สารบัญภาพ.....	ฎ
บทที่ 1.....	14
บทนำ.....	14
ความเป็นมาและความสำคัญของปัญหา.....	14
จุดมุ่งหมายของการศึกษา.....	15
ความสำคัญของการศึกษา.....	15
ขอบเขตของงานวิจัย.....	15
ข้อตกลงเบื้องต้น.....	16
สมมติฐานของการวิจัย.....	17
บทที่ 2.....	18
เอกสารและงานวิจัยที่เกี่ยวข้อง.....	18
ทฤษฎีที่เกี่ยวข้อง.....	18
1. มัลแวร์ (Malware).....	18
2. ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบเดิม.....	22

3. ความแตกต่างของระบบเปิดกับระบบกึ่งปิด	23
4. ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบใหม่.....	26
5. หลักการ CPS	27
6. SSH โพรโตคอล.....	28
7. SFTP โพรโตคอล.....	29
8. Paramiko library.....	29
9. โปรแกรม 7-zib.....	30
10. การเข้ารหัสลับ AES	31
11. โปรแกรม Snort.....	32
12. Firewall.....	33
13. Raspberry Pi	36
14. Tkinter	38
บทที่ 3.....	41
วิธีดำเนินงานวิจัย.....	41
1. กำหนดปัญหา.....	41
2. ศึกษาเอกสารและงานวิจัยที่เกี่ยวข้อง.....	41
3. วิเคราะห์และออกแบบสถาปัตยกรรมระบบ	43
4. ขั้นตอนการติดตั้งและการทำงานของระบบ.....	45
5. ขนาดไฟล์ที่ใช้ในการทดลอง.....	51
6. ขั้นตอนการทดลอง	53
7. ข้อมูลด้านฮาร์ดแวร์และระบบปฏิบัติการ.....	53
8. การประเมินและวัดผลการทดลอง.....	59

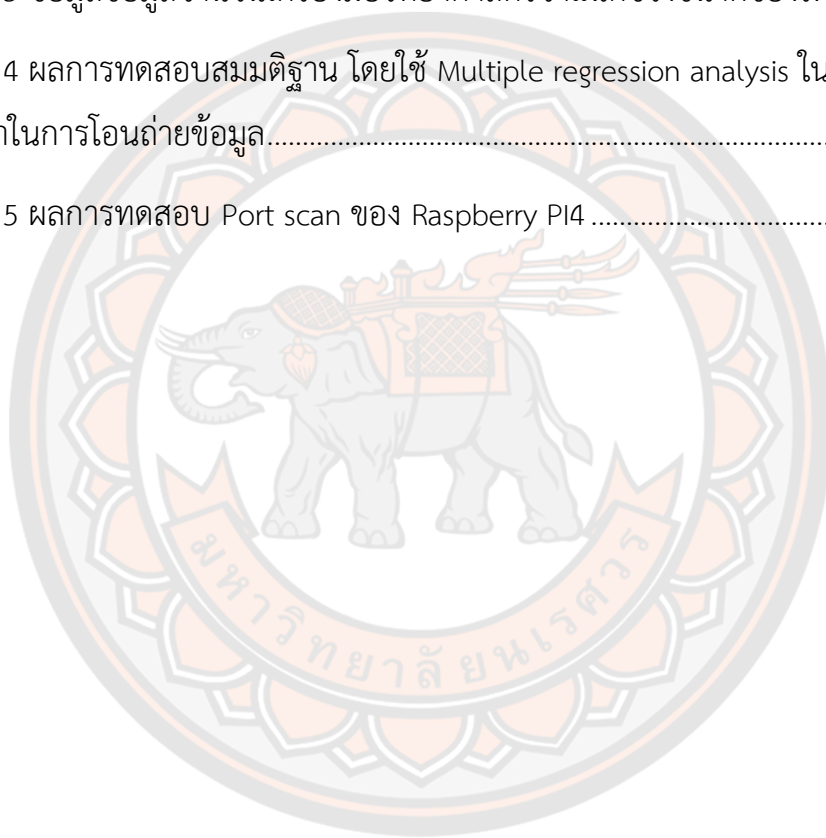
บทที่ 4.....	60
ผลการวิจัย.....	60
1. ผลการทดลองหาความสัมพันธ์ขนาดไฟล์กับเวลาในการโอนถ่ายข้อมูลให้สอดคล้องกับ การตั้งเวลาเปิดปิด Interface network ของ Raspberry Pi4 เชื่อมต่อกับ File server	60
สัญลักษณ์ที่ใช้ในการวิเคราะห์ข้อมูล	60
สมมติฐานข้อที่ 1 เวลาในการเปิด Interface network ของ Raspberry Pi4 เชื่อมต่อกับ File server มีค่าเฉลี่ยเวลาในการโอนถ่ายข้อมูลแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ที่ระดับนัยสำคัญ 0.05	62
สมมติฐานข้อที่ 2 ปริมาณของจำนวนไฟล์ที่แตกต่างกันใช้ระยะเวลาโอนถ่ายข้อมูล แตกต่างกันอย่างมีนัยสำคัญทางสถิติ ที่ระดับนัยสำคัญ 0.05	62
สมมติฐานข้อที่ 3 เวลาในการเปิด Interface network ของ Raspberry Pi4 ขนาดไฟล์ และจำนวนข้อมูลมีผลต่อระยะเวลาโอนถ่ายข้อมูลอย่างมีนัยสำคัญทางสถิติ ที่ระดับ นัยสำคัญ 0.05	62
การทดลองนี้เป็นการทดลองแสดงให้เห็นถึงความสัมพันธ์ของ เวลาเปิด ปิด Interface network ของ Raspberry Pi4 กับขนาดไฟล์ที่ใช้ในการแบ่งข้อมูลผลการวิเคราะห์ เพื่อใช้เป็นแนวทางในการหาเวลาเปิดปิด Interface network ที่เหมาะสมกับขนาด ไฟล์ที่จะแบ่งที่ทำเวลาถ่ายโอนข้อมูลเฉลี่ยที่ดีที่สุด.....	63
เมื่อนำผลเวลาเฉลี่ยการถ่ายโอนข้อมูลมาพล็อตกราฟแยกตามเวลาการเปิดปิด Interface network ของ Raspberry Pi4 เพื่อดูความสัมพันธ์ขนาดไฟล์กับเวลาใน การโอนถ่ายข้อมูลสอดคล้องกับการตั้งเวลาเปิดปิดที่ทำเวลาได้ดีที่สุดดังภาพ 36.64	
2. ผลการทดลองความถูกต้องของข้อมูลผลการวิเคราะห์ที่ได้รับผ่านทาง website.....	70
บทที่ 5.....	77
สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ.....	77

1.สรุปผลการวิจัย	77
2. อภิปรายผล.....	78
3.ข้อเสนอแนะ	79
บรรณานุกรม.....	81
ประวัติผู้วิจัย	84



สารบัญตาราง

	หน้า
ตาราง 1 เปรียบเทียบข้อดีข้อเสียของ Firewall แต่ละประเภท (kankann, 2563).....	35
ตาราง 2 ข้อมูลผลการวิเคราะห์ด้วยเครื่องมือวิทยาศาสตร์ 6 ปีย้อนหลัง.....	51
ตาราง 3 ข้อมูลข้อมูลจำนวนเครื่องมือวิทยาศาสตร์จำแนกช่วงขนาดของไฟล์.....	52
ตาราง 4 ผลการทดสอบสมมติฐาน โดยใช้ Multiple regression analysis ในการวิเคราะห์ หาเวลาในการโอนถ่ายข้อมูล.....	61
ตาราง 5 ผลการทดสอบ Port scan ของ Raspberry PI4	66



สารบัญภาพ

	หน้า
ภาพ 1 Predicted damage cost of Ransomware attack (Humayun et al., 2021).....	19
ภาพ 2 Taxonomy of Ransomware attack (Humayun et al., 2021)	19
ภาพ 3 How crypto Ransomware Work.(Humayun et al., 2021).....	20
ภาพ 4 How locker Ransomware Work.(Humayun et al., 2021).....	20
ภาพ 5 สถิติภัยคุกคามทางไซเบอร์ 3 ปีย้อนหลังจำแนกตามหน่วยงาน (NCSA, 2567)	21
ภาพ 6 สถิติภัยคุกคามทางไซเบอร์ 3 ปีย้อนหลังจำแนกตามการโจมตี (NCSA, 2567).....	22
ภาพ 7 ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบเดิม	23
ภาพ 8 ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบระบบเปิด	24
ภาพ 9 ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบระบบกึ่งปิด	25
ภาพ 10 ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบใหม่.....	26
ภาพ 11 ขั้นตอนการทำงานของโปรโตคอล SSH (SSH.COM, 2021c).....	29
ภาพ 12 ตัวอย่างการใช้ Paramiko library โอนถ่ายข้อมูลไป Raspberry Pi	30
ภาพ 13 ตาราง download โปรแกรม 7-zip (7-zip download, 2021).....	31
ภาพ 14 ตัวอย่างการใช้งานโปรแกรม 7-zip.....	31
ภาพ 15 Snort Architecture (Gogoi, 2018)	33
ภาพ 16 Raspberry Pi4 model B (Raspberrypi.org, 2021)	36
ภาพ 17 Designed system architecture (Karahana & Berat, 2020).....	37
ภาพ 18 Intrusion Detection Systems (Karahana & Berat, 2020).....	37
ภาพ 19 ตัวอย่างการสร้าง Root, Content frame และ Widgets.....	39

ภาพ 20 สถาปัตยกรรมของระบบ	43
ภาพ 21 ขั้นตอนการติดตั้งระบบถ่ายโอนข้อมูลแบบกึ่งปิด	46
ภาพ 22 โปรแกรม Nano data transfer	47
ภาพ 23 โปรแกรม process_send_file.exe	47
ภาพ 24 ตัวอย่างการบีบอัดและแบ่งไฟล์ด้วยโปรแกรม 7zip ที่ 550MB.....	48
ภาพ 25 ภาพผลตัวอย่างการบีบอัดและแบ่งไฟล์ด้วยโปรแกรม 7zip ที่ 100MB.....	48
ภาพ 26 ตัวอย่างการเข้ารหัสลับแบบ AES ด้วย pyAesCrypt	49
ภาพ 27 ผลตัวอย่างการเข้ารหัสลับแบบ AES ด้วย pyAesCrypt.....	49
ภาพ 28 โปรแกรม send_fileto_raspberry.exe	50
ภาพ 29 ตัวอย่างการเขียนไฟล์เมื่อถ่ายโอนไฟล์สำเร็จ	50
ภาพ 30 ความเร็วโอนถ่ายไฟล์ข้อมูลจาก File server ไป Raspberry Pi4.....	54
ภาพ 31 เป็นกราฟแสดงเวลาโอนถ่ายข้อมูล 1000MB 20 ครั้ง จาก File server ไป Raspberry Pi4	55
ภาพ 32 กราฟแสดงเวลาโอนถ่ายข้อมูลจาก USB Gigabit Ethernet adapter ของ File server ไปยัง Local Ethernet adapter ของ Raspberry Pi4.....	56
ภาพ 33 กราฟแสดงเวลาโอนถ่ายข้อมูลจาก USB Gigabit Ethernet adapter ของ File server ไปยัง USB Gigabit Ethernet adapter ของ Raspberry Pi4	57
ภาพ 34 กราฟแสดงเวลาโอนถ่ายข้อมูลจาก Local Ethernet adapter ของ File server ไปยัง Local Ethernet adapter ของ Raspberry Pi4.....	57
ภาพ 35 กราฟแสดงเวลาโอนถ่ายข้อมูลจาก File server ไปยัง Raspberry Pi4	58
ภาพ 36 แผนภูมิแสดงเวลารวมการโอนถ่ายข้อมูลเฉลี่ยเทียบกับขนาดการแบ่งไฟล์แยกตามเวลาเปิดปิด Interface ของ Raspberry Pi4	64

ภาพ 37 แผนภูมิแสดงเวลารวมการโอนถ่ายข้อมูลเฉลี่ยเทียบกับขนาดการแบ่งไฟล์แยกตามเวลาเปิดปิด Interface ของ Raspberry PI4	65
ภาพ 38 ทดสอบ Scan port 22.....	67
ภาพ 39 Snort detect scan port 22	67
ภาพ 40 Ping packets 1000 packet.....	68
ภาพ 41 Snort detect ICMP	68
ภาพ 42 Snort log detect ICMP	69
ภาพ 43 แสดง log firewall block scan port 22.....	70
ภาพ 44 แสดงลำดับการถ่ายโอนและการตรวจสอบความถูกต้องของข้อมูล	70
ภาพ 45 แสดงตัวอย่างการตั้งชื่อไฟล์ และ Folders	72
ภาพ 46 แสดงโปรแกรม Nano data transfer	72
ภาพ 47 แสดงขนาด Folders จำนวนไฟล์และจำนวน Folders ก่อนส่งไปยัง File server	73
ภาพ 48 แสดงรายการจองเครื่องมือวิทยาศาสตร์ทั้งหมดที่นักวิทยาศาสตร์รับผิดชอบดูแล	74
ภาพ 49 แสดงรายการจองเครื่องมือวิทยาศาสตร์ทั้งหมดที่นักวิทยาศาสตร์รับผิดชอบดูแลที่มีข้อมูลผลการวิเคราะห์	74
ภาพ 50 แสดงไฟล์ที่โหลดจากระบบ และผลจากการ unzip ไฟล์	75
ภาพ 51 แสดงขนาด Folders จำนวนไฟล์และจำนวน Folders ที่ได้จากระบบ	75
ภาพ 52 แสดงรายการจองเครื่องมือวิทยาศาสตร์ทั้งหมดของผู้รับบริการที่มีข้อมูลผลการวิเคราะห์.....	76
ภาพ 53 แสดงการกระจายอินเทอร์รัปต์บน CPU ทั้ง 4 ตัวบน Raspberry PI4.....	80

บทที่ 1

บทนำ

ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันความก้าวหน้าทางวิทยาศาสตร์และเทคโนโลยีเป็นปัจจัยสำคัญต่อคุณภาพชีวิตของทุกคน เครื่องมือต่าง ๆ ของใช้ในชีวิตประจำวันตลอดจนถึงยารักษาโรคล้วนต้องใช้เทคโนโลยีขั้นสูงในการผลิตตรวจสอบวิเคราะห์ประสิทธิภาพเพื่อให้ได้ผลผลิตที่เป็นที่ยอมรับและมีความปลอดภัยตลอดจนต้องค้ำความรู้ใหม่ ๆ

เครื่องมือวิเคราะห์ทางวิทยาศาสตร์มีส่วนสำคัญอย่างมากช่วยสนับสนุนการศึกษาให้เกิดองค์ความรู้ใหม่ ๆ พัฒนาเทคโนโลยี ใหม่ ๆ พร้อมกันนั้นเทคโนโลยีก็มีส่วนช่วยให้เกิดการพัฒนาระบบเครื่องมือวิเคราะห์ให้มีประสิทธิภาพโดยใช้เทคโนโลยีขั้นสูงควบคุมการทำงานด้วยคอมพิวเตอร์ร่วมกับระบบปฏิบัติการและโปรแกรมเฉพาะทาง การใช้คอมพิวเตอร์ในการควบคุมมีข้อดีหลายอย่างเช่นสามารถทำให้เครื่องมือทำงานได้มากและไวขึ้น มีความถูกต้องแม่นยำมากขึ้น แต่บางครั้งก็ทำให้เกิดปัญหากับผู้ใช้งานโดยเฉพาะกับผู้ที่ไม่มีความรู้พื้นฐานทางคอมพิวเตอร์เมื่อเครื่องมือมีปัญหาที่ไม่รู้ว่าเป็นข้อบกพร่องของเครื่องมือ หรือคอมพิวเตอร์ที่ควบคุมการทำงาน ปัญหาที่สำคัญมากในการใช้คอมพิวเตอร์ควบคุมและประมวลผลคือ Malware และไวรัสคอมพิวเตอร์ ถ้าเครื่องคอมพิวเตอร์ติด Malware หรือไวรัส จะส่งผลให้เครื่องคอมพิวเตอร์ทำงานผิดพลาดหรือไม่สามารถทำงานได้ แนวทางแก้ไขมีทางเดียวคือต้องให้ทางวิศวกรของบริษัทที่จำหน่ายและผลิตเครื่องมือนั้นเดินทางมาแก้ไขโดยการลงระบบปฏิบัติการและโปรแกรมเฉพาะทางในการควบคุมนั้นใหม่เท่านั้น และในการลงระบบปฏิบัติการและโปรแกรมเฉพาะทางใหม่จะมีค่าใช้จ่าย License ที่มีราคาสูง ดังนั้นหน่วยงานที่มีเครื่องมือวิเคราะห์ทางวิทยาศาสตร์ขั้นสูงไว้ให้บริการโดยเฉพาะเครื่องมือวิทยาศาสตร์สำหรับวิเคราะห์ตัวอย่างชิ้นงานของคุณะวิทยาศาสตร์มหาวิทยาลัยนครสวรรค์ ได้นำไปใช้เพื่อการเรียนการสอน และเป็นศูนย์เครื่องมือวิทยาศาสตร์ตั้งอยู่ที่คณะวิทยาศาสตร์ มหาวิทยาลัยนครสวรรค์ ให้บริการวิเคราะห์ตัวอย่างให้กับนิสิต บุคลากร ภายในและภายนอกมหาวิทยาลัย จึงมีมาตรการและแนวปฏิบัติออกมาเพื่อป้องกัน เช่นห้ามเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิเคราะห์ติดต่อกับเครือข่าย Internet และห้ามนำอุปกรณ์ภายนอกมาเชื่อมต่อกับเครื่องเช่น Flash drive, External hard drive โดยให้ใช้ได้แต่ CD หรือ DVD เท่านั้น แต่การใช้งาน CD หรือ DVD ก็ยังเกิดความไม่ปลอดภัยถ้า CD หรือ DVD ถูกเขียนมาก่อนหน้าที่จะนำมาใส่ในเครื่องคอมพิวเตอร์ และมาตรการดังกล่าวยังเกิดความไม่สะดวกกับผู้ใช้งานและผู้รับบริการและในปัจจุบัน CD หรือ DVD เริ่มหายากเพราะถูกแทนที่ด้วยสื่ออื่นและในอนาคตอาจต้องยุติการผลิต

ดังนั้นผู้วิจัยจึงได้ทำการพัฒนาระบบถ่ายโอนข้อมูลแบบกึ่งปิด (Semi-closed Data Transfer System) เพื่อแก้ไขปัญหาการนำผลการวิเคราะห์ตัวอย่างชิ้นงานออกจากเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิเคราะห์ และป้องกันการโจมตีจาก Malware ไวรัสคอมพิวเตอร์ และจากผู้ไม่ประสงค์ดี ซึ่งงานวิจัยนี้มุ่งเน้นความปลอดภัยของเครื่องคอมพิวเตอร์ ควบคุมการทำงานของเครื่องมือวิเคราะห์ ข้อมูลผลการวิเคราะห์ตัวอย่างและอำนวยความสะดวกให้กับนักวิทยาศาสตร์ นักวิจัยและผู้นำตัวอย่างมาวิเคราะห์อย่างเป็นระบบ

จุดมุ่งหมายของการศึกษา

1. ได้ระบบถ่ายโอนข้อมูลแบบกึ่งปิดที่นำข้อมูลขึ้นบนอินเทอร์เน็ตได้
2. ระบบจะหลีกเลี่ยงการเพิ่มช่องโหว่ทางด้านความปลอดภัยต่อเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิเคราะห์ และข้อมูลผลการวิเคราะห์
3. ระบบให้บริการข้อมูลผลการวิเคราะห์ผ่านทาง website

ความสำคัญของการศึกษา

1. ได้ระบบถ่ายโอนข้อมูลแบบกึ่งปิดที่สามารถนำข้อมูลผลการวิเคราะห์ตัวอย่างชิ้นบนอินเทอร์เน็ตส่งต่อข้อมูลให้ผู้รับบริการผ่านทางเว็บไซต์ได้อย่างถูกต้องและปลอดภัย
2. ระบบใช้งานได้จริง มีความปลอดภัยต่อเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิเคราะห์ และข้อมูลผลการวิเคราะห์

ขอบเขตของงานวิจัย

ด้านข้อมูล

1. ข้อมูลที่เป็น Text
2. ข้อมูลที่เป็นรูปภาพ
3. ข้อมูลสถิติการใช้งาน และความพึงพอใจของผู้ใช้งานจริง

ด้านระบบ

1. ระบบสามารถส่งข้อมูลผลการวิเคราะห์ตัวอย่างจากเครื่องคอมพิวเตอร์ในศูนย์เครื่องมือวิทยาศาสตร์ที่ใช้ระบบปฏิบัติการเป็น windows เท่านั้น
2. ระบบสามารถตรวจสอบและป้องกันการโจมตีที่มีต่อระบบฯเข้าเท่านั้น
3. ระบบบริการข้อมูลจะถูกพัฒนารองรับการใช้งานกับศูนย์เครื่องมือของคณะวิทยาศาสตร์ มหาวิทยาลัยนครสวรรค์เท่านั้น

ด้านเทคโนโลยี

1. หลักการ CPS ใช้ออกแบบระบบ
 - 1.1. ใช้ Raspberry Pi4 เป็นตัวกลางโอนถ่ายข้อมูลทำหน้าที่ เปิด – ปิด อินเทอร์เน็ต Network และตรวจสอบป้องกันการโจมตี
 - 1.2. ใช้ Firewall ในการควบคุมการทำงานของ Port การเชื่อมต่อและกรอง Packet ที่จะเข้ามาในระบบ
 - 1.3. ใช้ Snort เป็นเครื่องมือตรวจจับการบุกรุกทางเครือข่าย (Network intrusion detection)
2. การเข้ารหัสลับ AES ใช้เข้ารหัสข้อมูลผลการวิเคราะห์
3. ภาษา Python ใช้ควบคุมการทำงานของระบบ
 - 3.1. ใช้ PostgreSQL เป็นฐานข้อมูล
 - 3.2. ใช้ Paramiko library โอนถ่ายข้อมูลโดยใช้โปรโตคอล SFTP
 - 3.3. ใช้ Program 7zip จัดการไฟล์ที่มีขนาดใหญ่
 - 3.4. Tkinter สำหรับการพัฒนา GUI ในภาษา python
 - 3.5. ใช้ pycrypto library เข้ารหัสลับแบบ AES
4. ภาษา PHP และฐานข้อมูล MySQL ใช้จัดการข้อมูลผลการวิเคราะห์

ข้อตกลงเบื้องต้น

ได้ระบบถ่ายโอนข้อมูลแบบกึ่งปิดที่สามารถนำข้อมูลผลการวิเคราะห์ตัวอย่างขึ้นบนอินเทอร์เน็ตส่งต่อข้อมูลให้ผู้รับบริการได้อย่างถูกต้องและระบบสามารถแจ้งเตือนป้องกันการโจมตีจากภายนอกได้

นิยามศัพท์เฉพาะ

SSH (Secure Shell) หมายถึงโปรโตคอลที่ใช้ในการติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์บนระบบเครือข่ายผ่านพอร์ตหมายเลข 22

SFTP (Secured File Transfer Protocol) หมายถึงโปรโตคอลติดต่อสื่อสารระยะไกลระหว่าง Client และ Server เพื่อให้ผู้ใช้งานเข้าถึงเอกสาร จัดการเอกสาร เคลื่อนย้ายเอกสาร ได้อย่างปลอดภัย

Paramiko หมายถึง SSH library บน Python สำหรับ SSH ไปหาอุปกรณ์ปลายทาง

Firewall หมายถึงซอฟต์แวร์หรือฮาร์ดแวร์ในระบบเครือข่าย ทำหน้าที่เป็นตัวกรองข้อมูลสื่อสาร

CPS (Cyber physical systems) หมายถึงระบบไซเบอร์-กายภาพ เป็นการเชื่อมต่อทางวิศวกรรมที่บูรณาการโลกกายภาพ (Physical World) กับโลกไซเบอร์ (Cyber World) เข้าด้วยกัน

AES (Advanced Encryption Standard) หมายถึงมาตรฐานการเข้ารหัสลับข้อมูลอิเล็กทรอนิกส์ที่สร้างขึ้นโดยสถาบันมาตรฐานและเทคโนโลยีแห่งชาติสหรัฐ (NIST) ในปี 2001

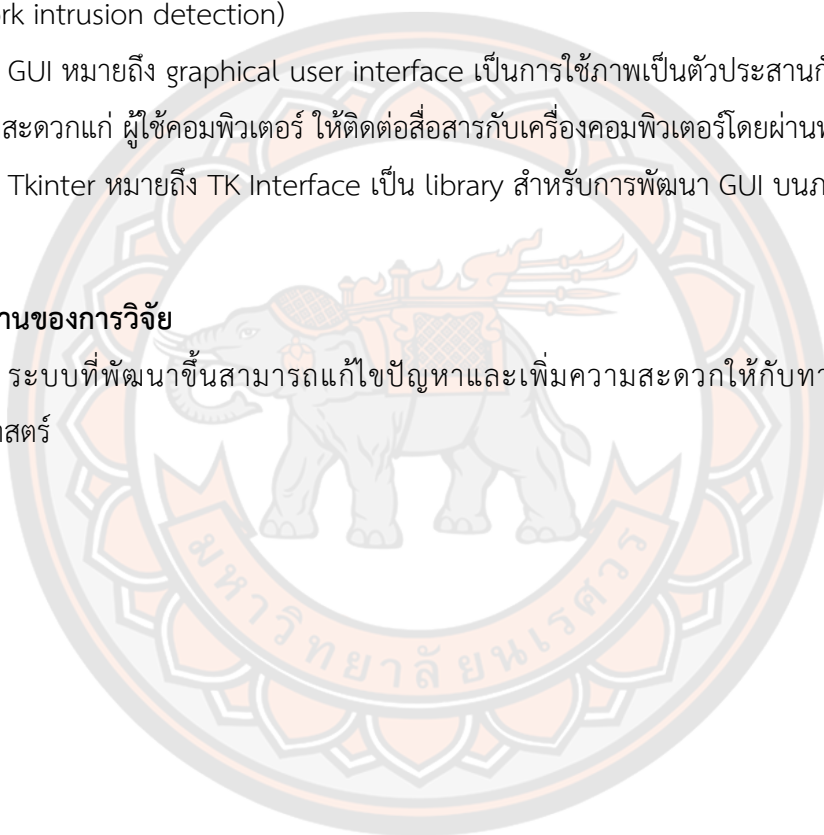
Snort หมายถึงโปรแกรม Open source เป็นเครื่องมือที่ใช้ตรวจจับการบุกรุกทางเครือข่าย (network intrusion detection)

GUI หมายถึง graphical user interface เป็นการใชภาพเป็นตัวประสานกับผู้ใช้ เป็นวิธีการให้ความสะดวกแก่ ผู้ใช้คอมพิวเตอร์ ให้ติดต่อสื่อสารกับเครื่องคอมพิวเตอร์โดยผ่านทางภาพ

Tkinter หมายถึง TK Interface เป็น library สำหรับการพัฒนา GUI บนภาษา python

สมมติฐานของการวิจัย

ระบบที่พัฒนาขึ้นสามารถแก้ไขปัญหาและเพิ่มความสะดวกให้กับทางศูนย์เครื่องมือวิทยาศาสตร์



บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้อง

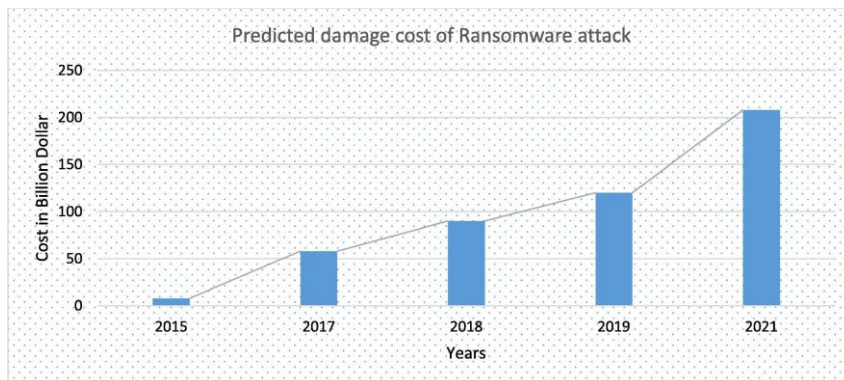
จากการศึกษาระบบการทำงานแบบเดิมของศูนย์เครื่องมือวิทยาศาสตร์และค้นคว้าเอกสารงานวิจัยที่เกี่ยวข้องเพื่อนำมาออกแบบบรรยากาศอินเทอร์เฟซข้อมูลแบบกึ่งปิดที่จะนำไปช่วยแก้ไขปัญหาให้กับทางศูนย์เครื่องมือวิทยาศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยนเรศวร ผู้ศึกษาค้นคว้าได้ศึกษาระบบการทำงานเดิมของทางศูนย์เครื่องมือวิทยาศาสตร์และเอกสารงานวิจัยที่เกี่ยวข้องดังนี้

1. มัลแวร์ (Malware)
2. ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบเดิม
3. ความแตกต่างของระบบเปิดกับระบบกึ่งปิด
4. ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบใหม่
5. หลักการ CPS
6. SSH โพรโตคอล
7. SFTP โพรโตคอล
8. Paramiko library
9. โปรแกรม 7zib
10. การเข้ารหัสลับ AES
11. โปรแกรม Snort
12. Firewall
13. Raspberry Pi
14. Tkinter

1. มัลแวร์ (Malware)

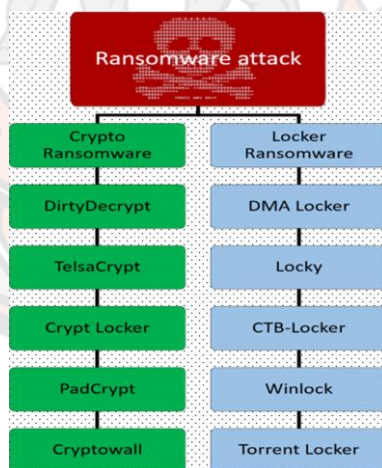
มัลแวร์ คือซอฟต์แวร์ที่เจตนาออกแบบมาเพื่อสร้างความเสียหายให้กับคอมพิวเตอร์ เซิร์ฟเวอร์ ไคลเอนต์ หรือเครือข่ายคอมพิวเตอร์ มัลแวร์มีหลากหลายประเภท รวมถึงไวรัสคอมพิวเตอร์ หนอนคอมพิวเตอร์ ม้าโทรจัน แรนซัมแวร์ สปายแวร์ แอดแวร์ และสแกร์แวร์ (Humayun et al., 2021) ได้นำเสนอ Internet of things and ransomware: Evolution, mitigation and prevention โดยมีวัตถุประสงค์เพื่อนักวิจัยและผู้ปฏิบัติงานได้ตระหนักถึงภัยคุกคามจาก Ransomware การป้องกัน Ransomware และการบรรเทาจากการโจมตีจาก Ransomware ใน IoT ผลที่ได้จาก

งานวิจัยนี้ ทำให้ทราบแนวโน้มค่าความเสียหายจาก Ransomware ที่จะมีแนวโน้มมากขึ้นเรื่อย ๆ ดังภาพ 1



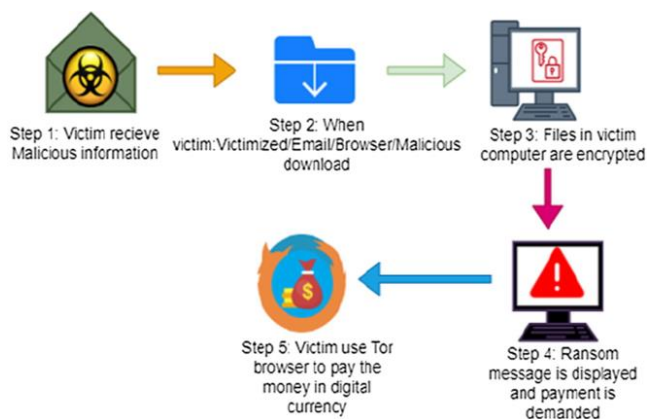
ภาพ 1 Predicted damage cost of Ransomware attack (Humayun et al., 2021)

ประเภทหลักของ Ransomware แบ่งออกเป็นสองประเภท คือ Crypto Ransomware และ locked Ransomware ดังภาพ 2



ภาพ 2 Taxonomy of Ransomware attack (Humayun et al., 2021)

ขั้นตอนการทำงานของ Ransomware ประเภท Crypto Ransomware ดังภาพ 3



ภาพ 3 How crypto Ransomware Work.(Humayun et al., 2021)

ขั้นตอนการทำงานของ Ransomware ประเภท Locker Ransomware ดังภาพ 4



ภาพ 4 How locker Ransomware Work.(Humayun et al., 2021)

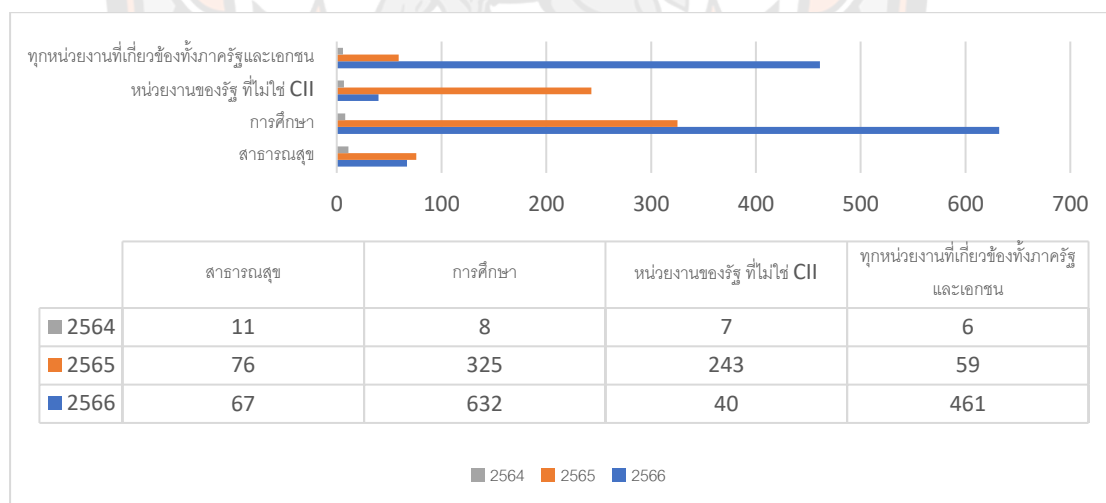
จากผลงานนี้ทำให้คาดการณ์ว่าการโจมตีของ Ransomware จะสูงขึ้น 5 เท่าในปี 2563 และจะเกิดความเสียหายมากกว่า 6 ล้านล้านดอลลาร์เนื่องจากค่าไถ่จากการโจมตีแรนซัมแวร์ งานวิจัยนี้มุ่งเน้นไปที่ IoT ที่มีการเชื่อมโยงการโจมตี ransomware วิธีการบรรเทาผลกระทบและการป้องกันที่แนะนำคือการฝึกอบรมให้กับผู้ใช้งานจะเป็นกุญแจสำคัญในการปกป้องอุตสาหกรรมองค์กรและบุคคลจากการติด malware และไวรัส งานวิจัยนี้ให้ข้อมูลเพื่อเป็นแนวทางการทำงานแก่นักวิจัยเพื่อป้องกันงานวิจัยไม่ให้ถูกโจมตีจาก Ransomware ในอนาคต

จากงานวิจัยนี้จะเห็นว่าผลกระทบของ Malware และไวรัสคอมพิวเตอร์มีผลกระทบอย่างมากทั่วโลกและเป็นสิ่งที่ใกล้ตัวมากที่สุดทำให้ผู้วิจัยต้องออกแบบระบบแบบกึ่งปิดแทนระบบแบบทั่วไปที่มีการเชื่อมต่อโดยตรงกับอินเทอร์เน็ตเพื่อความปลอดภัยของเครื่องคอมพิวเตอร์ควบคุมการ

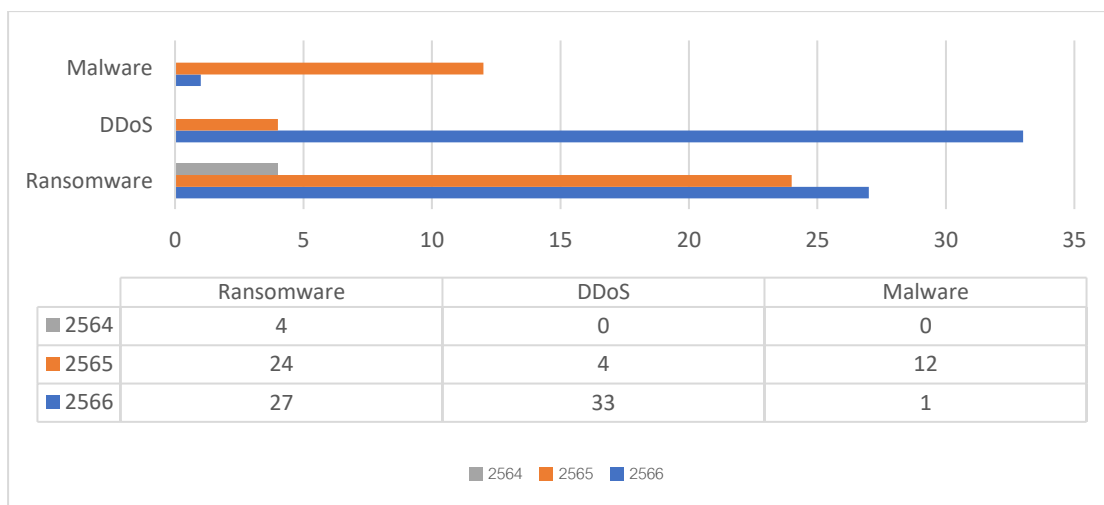
ทำงานของเครื่องมือวิเคราะห์ทางวิทยาศาสตร์ ที่ต้องตระหนักถึงอันตรายของ malware เช่น ransomware เพราะ

1. เครื่องมือทางวิทยาศาสตร์ มีราคาแพง และมีค่าบำรุงรักษาที่สูง
2. Software ที่ใช้ทำงานกับเครื่องมือส่วนใหญ่ทำงานบนปฏิบัติการ Windows ที่เก่า และไม่มี การป้องกัน Malware และ Virus computer
3. จากสถิติผลงานวิจัยนี้พบว่าเป้าหมายหลักของ Malware เช่น Ransomware คือ ระบบปฏิบัติการ Windows เป็นส่วนใหญ่
4. สิ่งที่ทางศูนย์เครื่องมือวิทยาศาสตร์และภาควิชามีความกังวลคือ กลัวเครื่องคอมพิวเตอร์ที่ ติดตั้ง Software สำหรับวิเคราะห์ ติด Malware และ Virus computer ถ้าเครื่องติดแล้ว ร้ายแรงสุดคือเครื่องนั้นอาจใช้งานไม่ได้ จะต้องเสียเวลาและค่าใช้จ่ายที่แพงที่จะให้บริษัท ผู้ผลิตส่งวิศวกรมาแก้ไข Malware ที่น่ากลัวตัวหนึ่งก็คือ Ransomware

จากรายงานสถิติภัยคุกคามทางไซเบอร์ระหว่างปี 2564 – 2566 ของสำนักงานคณะกรรมการ การรักษาความมั่นคงปลอดภัยไซเบอร์แห่งชาติพบว่าหน่วยงานทางการศึกษาถูกคุกคามทางไซเบอร์ มากขึ้นทุกปี และยังคงโจมตีจาก Ransomware เพิ่มขึ้น ดังภาพ 5 และ 6



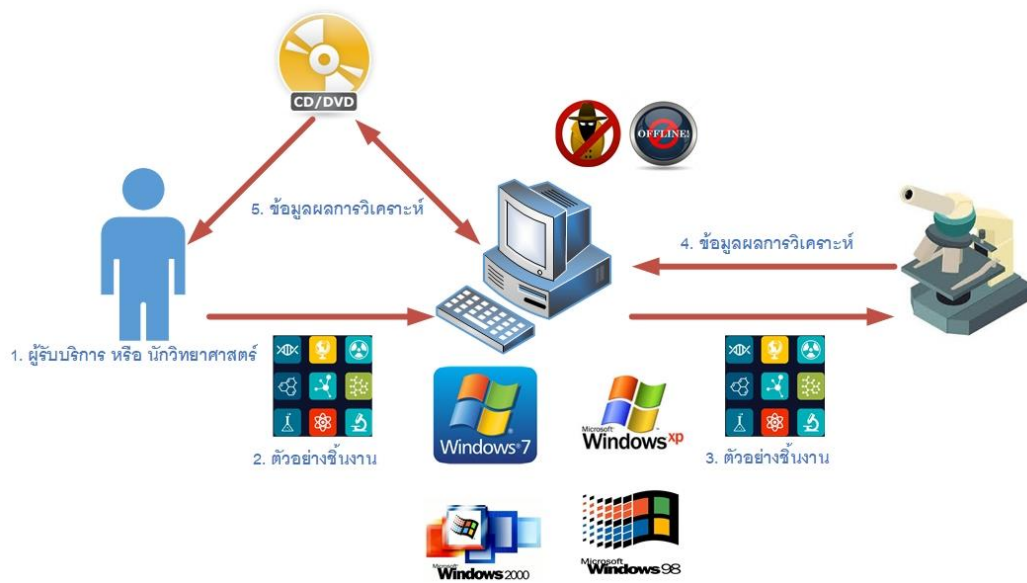
ภาพ 5 สถิติภัยคุกคามทางไซเบอร์ 3 ปีซ้อนหลังจำแนกตามหน่วยงาน (NCSA, 2567)



ภาพ 6 สถิติภัยคุกคามทางไซเบอร์ 3 ปีซ้อนหลังจำแนกตามการโจมตี (NCSA, 2567)

2. ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบเดิม

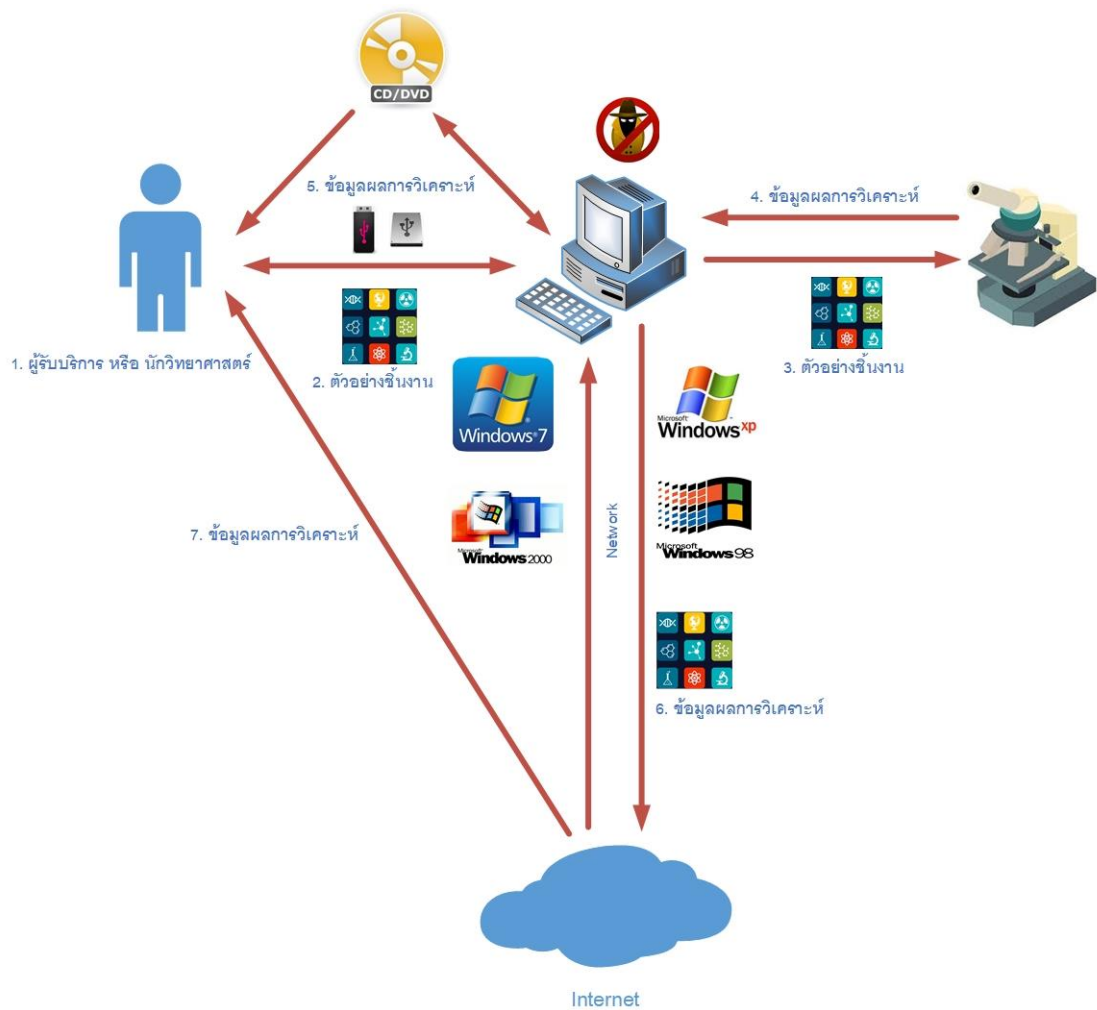
ขั้นตอนการใช้งานเครื่องมือวิทยาศาสตร์แบบเดิมของผู้รับบริการจะเริ่มจากการสมัครเป็นสมาชิกผ่านทางเว็บไซต์ของศูนย์เครื่องมือเมื่อผู้รับบริการต้องการใช้งานเครื่องมือวิเคราะห์ตัวอย่าง ชิ้นงานจะต้อง Login ผ่านเว็บไซต์ศูนย์เครื่องมือ และทำการจองเครื่องมือตามตารางเวลาของเครื่อง นั้น ๆ ที่นักวิทยาศาสตร์ได้เปิดคิวรับบริการไว้เมื่อจองการใช้งานเข้ามา นักวิทยาศาสตร์จะทำการตรวจรับการจอง การใช้งานเครื่องมือผู้รับบริการสามารถเดินทางเข้ามาใช้งานด้วยตนเองที่ศูนย์เครื่องมือ ภายใต้การแนะนำของนักวิทยาศาสตร์หรือจะให้นักวิทยาศาสตร์ทำการวิเคราะห์ตัวอย่างให้ เมื่อผู้รับบริการหรือนักวิทยาศาสตร์นำตัวอย่างชิ้นงานมาวิเคราะห์ผลการวิเคราะห์ที่ได้จะอยู่ภายในเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือ การนำผลการวิเคราะห์ออกจากคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือจะใช้วิธีเขียนลงแผ่น CD หรือ DVD เท่านั้น ดังภาพ 7



ภาพ 7 ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบเดิม

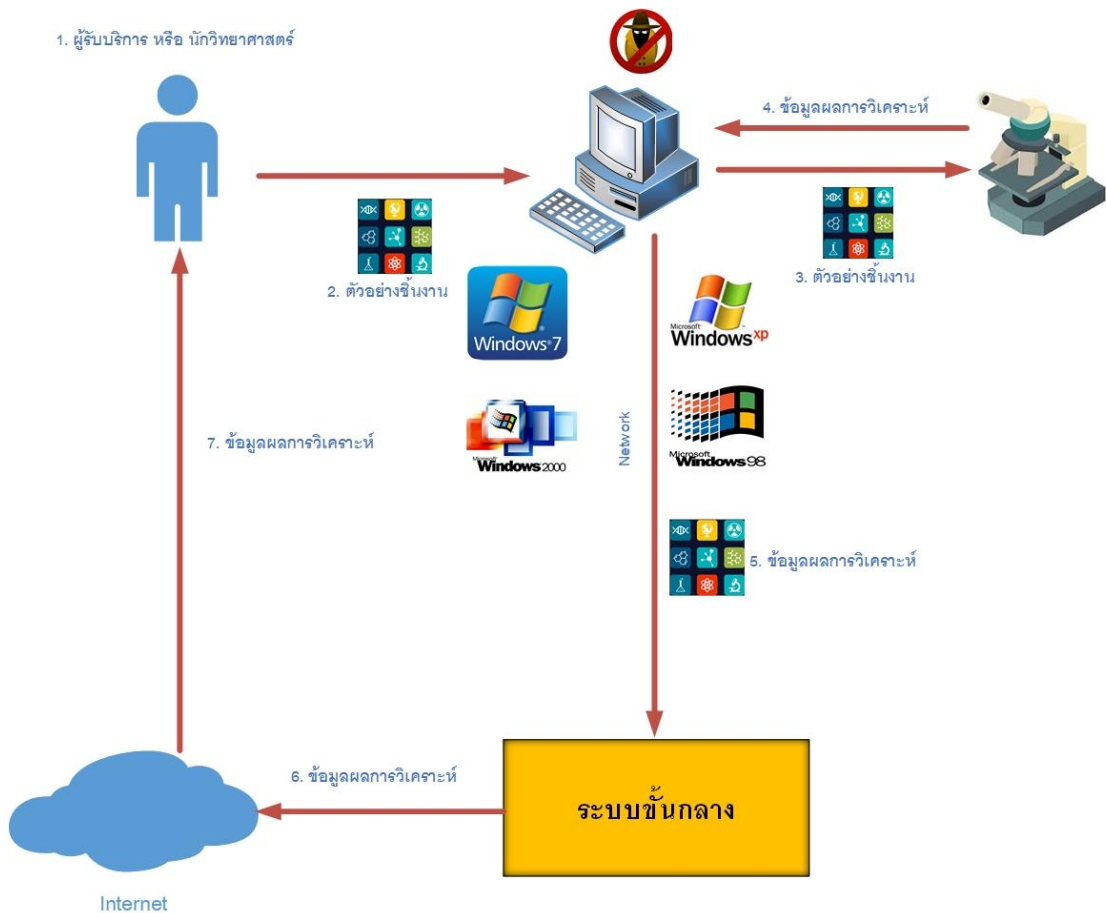
3. ความแตกต่างของระบบเปิดกับระบบกึ่งปิด

ระบบเปิดเป็นระบบที่ยอมให้คอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิเคราะห์เชื่อมต่อกับเครือข่ายเน็ตเวิร์คและอินเทอร์เน็ตเหมือนกับเครื่องคอมพิวเตอร์ใช้งานในสำนักงานทั่ว ๆ ไปโดยไม่มีอะไรมากั้นระหว่างเครื่องคอมพิวเตอร์กับระบบเครือข่ายมีการเชื่อมต่อกันอย่างอิสระเปิดโอกาสให้นักวิทยาศาสตร์หรือผู้รับบริการสามารถนำผลการวิเคราะห์ตัวอย่างออกจากเครื่องควบคุมการทำงานของเครื่องมือได้หลายช่องทางเช่นทางระบบเครือข่ายและนำอุปกรณ์ต่าง ๆ มาเชื่อมต่อเช่น Flash drive, External hard drive, CD และ DVD ดังภาพ 8 จากภาพ 8 สามารถอธิบายขั้นตอนการทำงานได้ดังนี้ ผู้รับบริการหรือนักวิทยาศาสตร์นำตัวอย่างชิ้นงานมาวิเคราะห์เมื่อทำการวิเคราะห์เสร็จข้อมูลผลการวิเคราะห์จะถูกจัดเก็บในเครื่องควบคุมการทำงานของเครื่องมือ ผู้รับบริการหรือนักวิทยาศาสตร์สามารถนำผลการวิเคราะห์ออกจากเครื่องควบคุมการทำงานของเครื่องมือผ่านระบบเครือข่ายเช่นการแชร์ข้อมูล นำข้อมูลไปเก็บไว้บนคลาวด์ นำข้อมูลบริการผ่านเว็บไซต์และนำเอาอุปกรณ์อื่น ๆ มาเชื่อมต่อนำข้อมูลออกไปใช้งาน



ภาพ 8 ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบระบบเปิด

ส่วนระบบแบบกึ่งปิดเครื่องคอมพิวเตอร์ควบคุมการทำงานจะไม่มี การเชื่อมต่อโดยตรงกับระบบเครือข่ายและอินเทอร์เน็ตโดยจะมีระบบหนึ่งระบบเข้ามาชั้นกลางคอยควบคุมการเปิดปิดอินเทอร์เน็ตเฟสเน็ตเวิร์คและมีระบบรักษาความปลอดภัย จะไม่มี การนำเอาอุปกรณ์ต่าง ๆ เข้ามาเชื่อมต่อเหมือนระบบแบบเปิดดังภาพ 9 จากภาพ 9 สามารถอธิบายขั้นตอนการทำงานได้ดังนี้ ผู้รับบริการหรือนักวิทยาศาสตร์นำตัวอย่างชิ้นงานมาวิเคราะห์เมื่อทำการวิเคราะห์เสร็จข้อมูลผลการวิเคราะห์จะถูกจัดเก็บในเครื่องควบคุมการทำงานของเครื่องมือและจะถูกส่งต่อให้ระบบชั้นกลางจัดการส่งต่อข้อมูลผลการวิเคราะห์ให้ผู้รับบริการอย่างเป็นระบบ

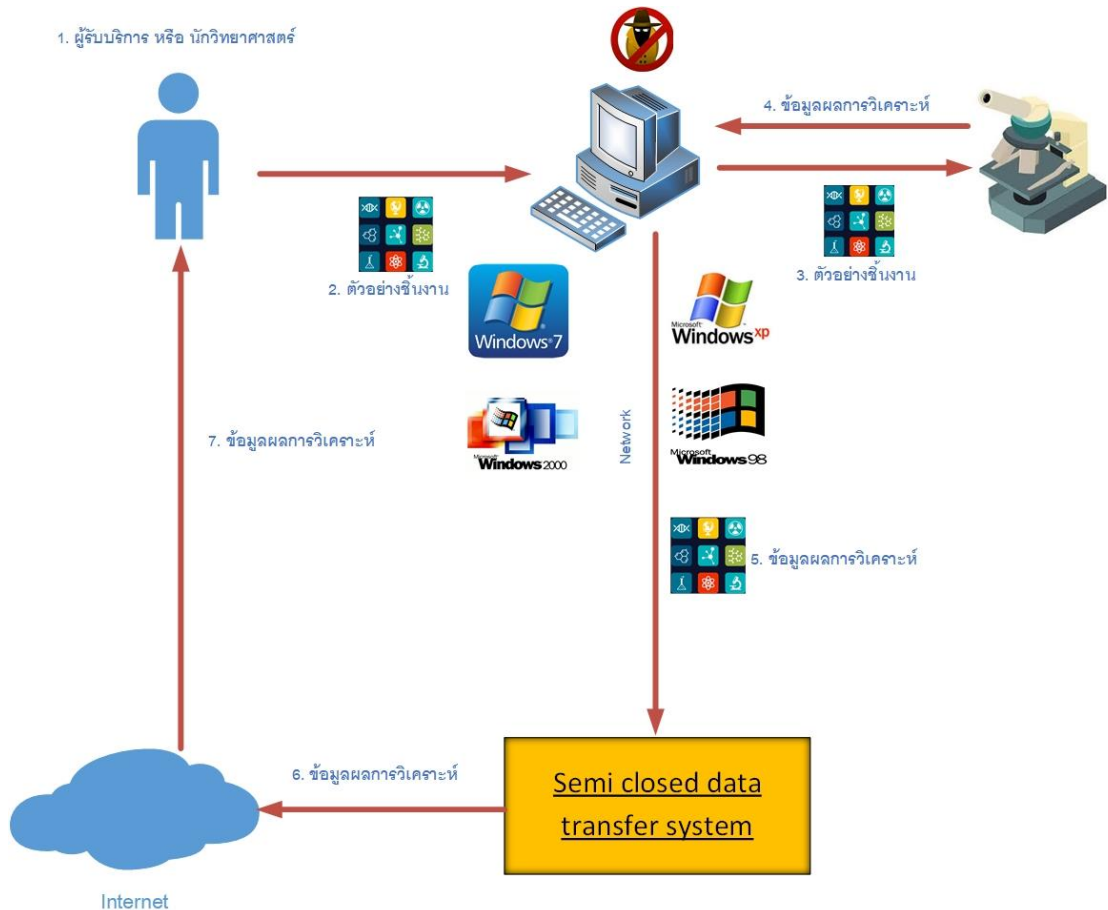


ภาพ 9 ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบระบบกึ่งปิด

จะเห็นว่าระบบแบบเปิดจะไม่มีความปลอดภัยในการนำเอาผลการวิเคราะห์ไปใช้งาน เนื่องจากเครื่องคอมพิวเตอร์ควบคุมการทำงานมีระบบปฏิบัติการ windows ที่เก่าและล้าสมัยและยังไม่มีโปรแกรมป้องกัน malware และไวรัสคอมพิวเตอร์เมื่อเชื่อมต่อเครือข่ายอินเทอร์เน็ตและนำเอาอุปกรณ์ต่าง ๆ มาเชื่อมต่อเครื่องคอมพิวเตอร์อาจถูกโจมตีจาก malware และไวรัสคอมพิวเตอร์และอาจถูกดักจับข้อมูลผลการวิเคราะห์จากผู้ไม่ประสงค์ดี ส่วนระบบแบบกึ่งปิดจะมีระบบชั้นกลางคอยบริหารจัดการและรักษาความปลอดภัยไม่ให้ถูกโจมตีจาก malware และไวรัสคอมพิวเตอร์ และปกป้องข้อมูลผลการวิเคราะห์จากผู้ไม่ประสงค์ดี

4. ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบใหม่

ระบบแบบใหม่ถูกพัฒนาขึ้นมาเพื่อแก้ไขปัญหาด้านความปลอดภัยของเครื่องคอมพิวเตอร์ ควบคุมการทำงานของเครื่องมือและปกป้องข้อมูลผลการวิเคราะห์และเพื่อเพิ่มความสะดวกให้นักวิทยาศาสตร์และผู้รับบริการนำออกข้อมูลผลการวิเคราะห์ได้รวดเร็ว



ภาพ 10 ขั้นตอนการใช้เครื่องมือวิทยาศาสตร์แบบใหม่

จากภาพที่ 10 สามารถอธิบายขั้นตอนได้ดังนี้เริ่มจากการสมัครเป็นสมาชิกผ่านทางเว็บไซต์ของศูนย์เครื่องมือเมื่อผู้รับบริการต้องการใช้งานเครื่องมือวิเคราะห์ตัวอย่างชิ้นงานจะต้อง Login ผ่านเว็บไซต์ศูนย์เครื่องมือ และทำการจองเครื่องมือตามตารางเวลาของเครื่องนั้นๆ ที่นักวิทยาศาสตร์ได้เปิดคิวรับบริการไว้เมื่อจองการใช้งานเข้ามานักวิทยาศาสตร์จะทำการตรวจรับการจอง การใช้งานเครื่องมือผู้รับบริการสามารถเดินทางเข้ามาใช้งานด้วยตนเองที่ศูนย์เครื่องมือภายใต้การแนะนำของนักวิทยาศาสตร์หรือจะให้นักวิทยาศาสตร์ทำการวิเคราะห์ตัวอย่างให้ เมื่อผู้รับบริการหรือนักวิทยาศาสตร์นำตัวอย่างชิ้นงานมาวิเคราะห์ผลการวิเคราะห์ที่ได้จะอยู่ภายในเครื่องคอมพิวเตอร์ ควบคุมการทำงานของเครื่องมือและจะถูกส่งต่อไปให้ระบบ ถ่ายโอนข้อมูลแบบกึ่งปิดระบบจะส่งต่อผล

การวิเคราะห์ให้ผู้รับบริการ Download ผ่านทางเว็บไซต์ โดยนักวิทยาศาสตร์และผู้รับบริการไม่ต้องนำเอาอุปกรณ์ต่าง ๆ มาเชื่อมต่อ

5. หลักการ CPS

(มีทรัพย์สินหลาย, 2562) ได้อธิบายหลักการ CPS เป็นการเชื่อมต่อทางวิศวกรรมที่บูรณาการโลกกายภาพ (Physical World) กับโลกไซเบอร์ (Cyber World) เข้าด้วยกัน โลกกายภาพประกอบด้วยสิ่งต่าง ๆ เช่น อุปกรณ์ เครื่องจักร มนุษย์ ระบบต่าง ๆ ที่มนุษย์สร้างขึ้นหรือเกิดขึ้นเองตามธรรมชาติ รวมถึงสภาพแวดล้อม ส่วนโลกไซเบอร์หรือโลกดิจิทัลเป็นโลกแห่งการประมวลผลและการควบคุมการผนวกสองโลกเข้าด้วยกันเริ่มจากการเชื่อมต่อของสิ่งต่าง ๆ ในโลกกายภาพเข้าเป็นเครือข่าย ซึ่งเทคโนโลยี Internet of Things (IoT) ก็เป็นตัวช่วยหนึ่งที่ทำให้เกิดการเชื่อมต่อ (Connectivity) การสื่อสาร (Communication) และการนำข้อมูลจากอุปกรณ์ เครื่องจักร หรือสถานะแวดล้อมต่าง ๆ ในโลกกายภาพส่งต่อไปให้โลกของไซเบอร์ช่วยประมวลผล (Computing) วิเคราะห์คำนวณ หรือตัดสินใจ เพื่อส่งข้อมูลย้อนกลับมาควบคุม (Feedback Control) โลกกายภาพอีกทีอย่างเป็นอัตโนมัติ เมื่อนำหลักการนี้มาใช้งานสิ่งที่ไม่ได้คือเรื่องของความปลอดภัยของข้อมูลและสิ่งต่าง ๆ ทางกายภาพที่นำมาเชื่อมต่อกันในเครือข่ายดังนั้นก็จะมีผลงานวิจัยที่ได้ทำการศึกษาเรื่องความปลอดภัยของ CPS (Ashibani & Mahmoud, 2017) ได้ทำการศึกษาเรื่อง Cyber physical systems security: Analysis, challenges and solutions จากผลงานวิจัยนี้ได้รับรู้โครงสร้างของระบบ Cyber Physical Systems ประกอบด้วยสาม Layer คือ

1. Application Layer ตัวอย่างเช่น Smart Home, Smart City, Smart Industry, Smart Building, Smart Transportation และ Smart Health เป็นต้น
2. Transmission Layer ตัวอย่างเช่น WI-FI, Bluetooth, Access Point, Router, The Internet และ Lan เป็นต้น
3. Perception Layer ตัวอย่างเช่น Sensors, RFID, Actuators และ GPS เป็นต้น

งานวิจัยนี้พบว่าการโจมตี CPS สามารถถูกโจมตีได้ทั้งสาม Layer ซึ่งงานวิจัยนี้ได้เกี่ยวข้องกับงานที่กำลังดำเนินการทำอยู่เพราะภาพรวมโครงสร้างของระบบถ่ายโอนข้อมูลแบบกึ่งปิดเป็นการนำเอาเครื่องมือวิเคราะห์ทางวิทยาศาสตร์และคอมพิวเตอร์ควบคุมการทำงานที่ไม่มีการเชื่อมต่อ Network ซึ่งเป็นโลกกายภาพเชื่อมต่อเข้ากับโลกไซเบอร์เพื่อนำออกไฟล์ข้อมูลผลการวิเคราะห์ไปยัง Web server เพื่อใช้งานซึ่งก็เป็นส่วนหนึ่งของหลักการ CPS แต่การนำเอาเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือต่าง ๆ เชื่อมต่อเข้ากับระบบ Network และ Internet สิ่งที่น่าสนใจคือความปลอดภัยของเครื่องคอมพิวเตอร์และข้อมูลที่จะถูกโจมตีจากผู้ไม่ประสงค์ดี งานวิจัยนี้ได้เสนอแนวทางการโจมตีและแนวทางป้องกันเพื่อใช้เป็นแนวทางการออกแบบระบบให้มีความปลอดภัยดังนี้

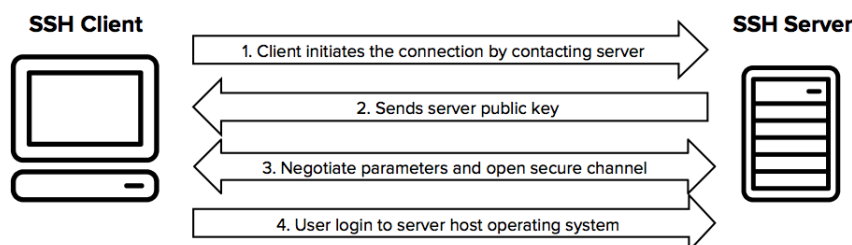
1. Perception Layer ได้แนะนำวิธีป้องกันโดย Data encryption, Secure routing protocol, Authentication
2. Transmission Layer ได้แนะนำวิธีป้องกันโดย Attack detection mechanism, Network access control, hop by hop data encryption
3. Application Layer ได้แนะนำวิธีป้องกันโดย End to end encryption, Intrusion detection, User authentication and authorization

นอกจากนี้ยังมีงานวิจัยที่ได้นำเอาหลักการ CPS มาประยุกต์ใช้ (Qiu et al., 2020) ได้ทำการศึกษาเรื่อง Secure Health Data Sharing for Medical Cyber-Physical Systems for the Healthcare 4.0 งานวิจัยนี้ประกอบด้วยองค์ประกอบหลักสองประการคือ (1) ออกแบบวิธีการจัดเก็บและแบ่งปันข้อมูลที่เน้นผู้ใช้เป็นศูนย์กลางใน MCPS บนระบบคลาวด์ที่มีความปลอดภัยและมีความเป็นส่วนตัวของข้อมูล EHR ของผู้ใช้ซึ่งสามารถปกป้องความปลอดภัยของข้อมูลและความเป็นส่วนตัวแม้ว่าเซิร์ฟเวอร์คลาวด์และคีย์จะถูกบุกรุก (2) ประเมินความเป็นไปได้ของระบบนี้ตาม Mobile Edge Computing (MEC) โดยใช้สมาร์ทโฟนพิสูจน์ปรับปรุงประสิทธิภาพเปรียบเทียบกับอัลกอริทึมการเข้ารหัสมาตรฐานจากการศึกษาวิจัยนี้สรุปได้ว่าการเข้ารหัสข้อมูลด้วยอัลกอริทึมสามารถหลีกเลี่ยงการรั่วไหลของข้อมูลได้แม้คีย์จะถูกบุกรุก งานวิจัยนี้เกี่ยวข้องกับงานที่ทำในส่วนของ การแชร์ข้อมูลจากเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือมือต่างๆผ่าน Network ไปยังคลาวด์หรือ Web server สำหรับให้บริการข้อมูลผลการวิเคราะห์ มีการป้องกันภัยคุกคามต่อความปลอดภัยของข้อมูลโดยการเข้ารหัสข้อมูลแต่ในส่วนของงานวิจัยนี้ไม่เหมือนกับงานที่จะทำคือมีการป้องกันการโจมตีของแหล่งข้อมูล

6. SSH โพรโทคอล

SSH โพรโทคอล (SSH.COM, 2021b) เป็นโพรโทคอลที่ใช้ในการติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์บนระบบเครือข่ายที่อนุญาตให้ผู้ใช้งานสามารถควบคุมหรือสั่งการเครื่องคอมพิวเตอร์เครื่องนั้น ๆ อย่างปลอดภัย โพรโทคอล SSH จะทำงานแบบ Client และ Server ประกอบไปด้วยโปรแกรม 2 ส่วนคือ โปรแกรมส่วนของ Server ซึ่งจะติดตั้งลงบน เครื่องที่ให้บริการ เป็นการเปิดช่องทางให้ผู้ใช้งานสามารถเข้ามาควบคุมการทำงานหรือสั่งการเครื่องที่ติดตั้งโปรแกรมนี้ได้เช่น OpenSSH และโปรแกรมส่วนของ Client ซึ่งจะติดตั้งลงบนเครื่อง Client หรือเครื่องคอมพิวเตอร์ที่จะทำหน้าที่เป็นผู้เชื่อมต่อไปยังเครื่องที่ให้บริการ เช่น Putty, WinSCP เป็นต้น OpenSSH เป็นโปรแกรม open source ที่ใช้โพรโทคอล SSH ในการติดต่อสื่อสารกันระหว่างหลายๆ application ไม่ว่าจะเป็นการใช้ remote file transfer อย่าง sftp, remote terminal, remote file copy อย่าง

scp ซึ่งการสื่อสารแบบนี้มีความปลอดภัยเพราะ SSH ใช้วิธีเข้ารหัสข้อมูลที่ส่งผ่านระหว่างเครื่องผ่านทางพอร์ต 22 ดังภาพที่ 11



ภาพ 11 ขั้นตอนการทำงานของโปรโตคอล SSH (SSH.COM, 2021c)

ในการออกแบบระบบถ่ายโอนข้อมูลแบบกึ่งปิดได้นำเอาโปรโตคอล SSH มาใช้ในการติดต่อสื่อสารระหว่าง File server กับ Raspberry Pi และ Raspberry Pi กับ Web server เพื่อเพิ่มประสิทธิภาพด้านความปลอดภัยในการติดต่อสื่อสาร

7. SFTP โปรโตคอล

SFTP (SSH Secure File Transfer Protocol) (SSH.COM, 2021a) เป็นโปรโตคอลที่ทำงานบน SSH โปรโตคอล นำมาใช้แทน FTP (File Transfer Protocol) ที่มีความปลอดภัยน้อย โดยมี sftp server เป็นโปรแกรมที่รันอยู่ที่ฝั่งเซิร์ฟเวอร์ รอรับการติดต่อจากไคลเอนต์ผ่านทางคำสั่ง sftp ทำหน้าที่ช่วยในการติดต่อสื่อสารระยะไกลระหว่าง Client-Server เพื่อให้ผู้ใช้งานเข้าถึงเอกสารจัดการเอกสาร เคลื่อนย้าย ได้ปลอดภัยมากยิ่งขึ้น ในการออกแบบระบบถ่ายโอนข้อมูลแบบกึ่งปิดได้นำเอาโปรโตคอล SFTP มาใช้ในการโอนถ่ายข้อมูลผลการวิเคราะห์ตัวอย่างระหว่าง File server ไปยัง Raspberry Pi และ Raspberry Pi ไปยัง Web server เพื่อเพิ่มประสิทธิภาพด้านความปลอดภัยในการโอนถ่ายข้อมูล

8. Paramiko library

Paramiko คือ SSH library บน Python (Forcier, 2021) ที่ค่อนข้างดีและมีความยืดหยุ่นสำหรับ SSH ไปหาอุปกรณ์ปลายทาง หรือจะใช้ transfer ไฟล์ด้วย SFTP (Khaokaew, 2563) มีฟังก์ชันการทำงานเบื้องต้นดังนี้

- set_missing_host_key_policy (paramiko.AutoAddPolicy()) เพิ่ม SSH key ไปยังไฟล์ known_hosts
- client.connect() เริ่ม connect ไปยัง SSH server
- client.close() ปิด SSH connection
- client.invoke_shell() สร้าง session
- session.send() ส่ง command ผ่าน session จาก invoke_shell()
- session.recv() รับ response จาก server ผ่าน session จาก invoke_shell()

ในการออกแบบระบบถ่ายโอนข้อมูลแบบกึ่งปิดได้นำเอา Paramiko library SFTP ส่งไฟล์จาก File server ไปยัง Raspberry Pi และส่งจาก Raspberry Pi ไปยัง Web server ดังภาพที่ 12

```

for row in range(len(lines)):
    fname = lines[row].strip()
    df = fname.split("@")
    print(df[1].strip())
    s = paramiko.SSHClient()
    s.set_missing_host_key_policy(paramiko.AutoAddPolicy())
    s.connect("10.31.10.34", 22, username='pi', password='1qaz2wsx3e', timeout=4)
    sftp = s.open_sftp()
    sftp.put('C://Temp_file/'+df[1]+'', 'Temp_File/'+fname+'', callback=byte_count, confirm=True)
    file = sftp.file('Temp_File/logfile.txt', "a", -1)
    file.write(fname+'\n')
    file.flush()
    listdel.append(fname)
    s.close()
text_file.close()

```

ภาพ 12 ตัวอย่างการใช้ Paramiko library โอนถ่ายข้อมูลไป Raspberry Pi

9. โปรแกรม 7-zib

โปรแกรม 7-Zip เป็นโปรแกรมสำหรับบีบอัดไฟล์ทำให้ไฟล์มีขนาดเล็กลง มีความสามารถบีบอัดไฟล์หลาย ๆ ไฟล์เข้าเป็นไฟล์เดียว เพื่อสะดวกในการคัดลอกจัดเก็บในอุปกรณ์เก็บข้อมูล หรือส่ง E-Mail และยังสามารถแบ่งไฟล์ออกเป็น part ตามขนาดที่ต้องการเหมาะสำหรับส่งไฟล์ขนาดใหญ่ในเครือข่ายเน็ตเวิร์ค โปรแกรมนี้มีหลักการทำงานเช่นเดียวกับโปรแกรมบีบอัดไฟล์ตามท้องตลาดทั่วไป คือ WinZIP และ WinRAR โปรแกรม 7-Zip เป็น Freeware สามารถทำงานกับไฟล์โดยใช้ฟังก์ชันบีบอัด (Add) และแตกไฟล์ (Extract) ไฟล์นามสกุล 7z, ZIP, GZIP, BZIP2 และ TAR และสามารถใส่ฟังก์ชันแตกไฟล์ (Extract) ได้อย่างเดียวกับไฟล์นามสกุล RAR, CAB, ISO, ARJ, LZH, CHM, Z, CPIO, RPM, DEB และ NSIS สามารถ Download โปรแกรมนี้ได้ที่ <https://www.7-zip.org/download.html> ในการออกแบบระบบถ่ายโอนข้อมูลแบบกึ่งปิดจะใช้งานโปรแกรม 7-zip

บีบอัดไฟล์ข้อมูลผลการวิเคราะห์และถ้าไฟล์ข้อมูลผลการวิเคราะห์มีขนาดใหญ่ให้ทำการแบ่งไฟล์ออกเป็น part เพื่อประสิทธิภาพในการโอนถ่ายข้อมูลทางเครือข่ายเน็ตเวิร์ค ในการเลือก Download โปรแกรม 7-zip มาใช้งานสำหรับ Windows x86 / x64 จะเลือกใช้ 7-Zip Extra: standalone console version, 7z DLL, Plugin for Far Manager ซึ่งสามารถใช้งานผ่านคำสั่งใน Command ได้

Download 7-Zip 19.00 (2019-02-21) for Windows:

Link	Type	Windows	Description
Download	.exe	32-bit x86	7-Zip for 32-bit Windows
Download	.exe	64-bit x64	7-Zip for 64-bit Windows x64 (Intel 64 or AMD64)
Download	.7z	x86 / x64	7-Zip Extra: standalone console version, 7z DLL, Plugin for Far Manager
Download	.7z	Any	7-Zip Source code
Download	.7z	Any / x86 / x64	LZMA SDK: (C, C++, C#, Java)
Download	.msi	32-bit x86	(alternative MSI installer) 7-Zip for 32-bit Windows
Download	.msi	64-bit x64	(alternative MSI installer) 7-Zip for 64-bit Windows x64 (Intel 64 or AMD64)

ภาพ 13 ตาราง download โปรแกรม 7-zip (7-zip download, 2021)

```

79     if event.event_type == 'created':
80         finish_copy = wait_for_file_copy_finish(r'+event.src_path)
81         if finish_copy == True:
82             x = x+1
83         try:
84             x = 1
85             paths = str(event.src_path)
86             head, tail = os.path.split(paths)
87             c = tail.split(".")
88             print(tail)
89             os.system("C://7z1900-extra/7za a -t7z C://Temp_file/"+c[0]+".7z E://File_book/"+tail+"")
90             st = os.path.getsize("c://Temp_file/"+c[0]+".7z")
91             print("-----"+str(st))
92             file = open(r"C://Temp_file/logfile.txt", "a")
93             file.write(str(st)+"@"+c[0]+".7z\n")

```

ภาพ 14 ตัวอย่างการใช้งานโปรแกรม 7-zip

10. การเข้ารหัสลับ AES

แนวทางการปกป้องข้อมูลในระบบเครือข่ายที่เป็นที่นิยมคือการเข้ารหัสลับข้อมูลเพื่อป้องกันข้อมูลไม่ให้ผู้ไม่ประสงค์ดีนำไปใช้งานได้ซึ่งได้มีผลงานวิจัยที่ใช้วิธีการปกป้องข้อมูล (ชัยพร ปานยินดี พุทธภรณ์ เอี่ยมภาณี นิษฐา อรุณสินประเสริฐ, 2560) ได้นำเสนอการรวมกันของสองขั้นตอนวิธี ประกอบด้วยการอำพรางข้อมูลแบบที่สามารถกู้คืนกลับได้ (Reversible Data Hiding: RDH) และ

การเข้ารหัสลับ (Advanced Encryption Standard: AES) เพื่อเพิ่มประสิทธิภาพความปลอดภัยในการเข้าถึงข้อมูล ในการศึกษาการรวมกันของวิทยาการเข้ารหัสลับกับวิทยาการเข้ารหัสลับสำหรับภาพทางการแพทย์ ผลลัพธ์สำหรับงานวิจัยนี้แสดงให้เห็นถึงความบิดเบือนที่ต่ำสำหรับประสิทธิภาพในการอำพรางและความปลอดภัยที่สูงขึ้นสำหรับการเข้าถึงข้อมูล ในการออกแบบระบบถ่ายโอนข้อมูลแบบกึ่งปิดได้นำวิธีการเข้ารหัสลับมาประยุกต์ใช้งานเพื่อเพิ่มประสิทธิภาพความปลอดภัยในการเข้าถึงข้อมูลผลการวิเคราะห์ตัวอย่างโดยการเข้ารหัสลับ Advanced Encryption Standard: AES

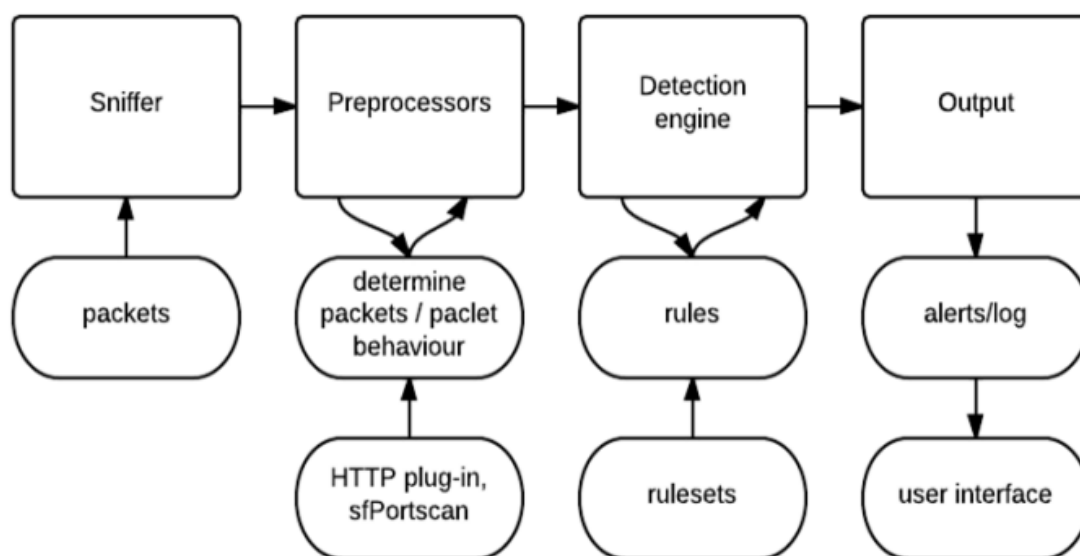
11. โปรแกรม Snort

Snort เป็นโปรแกรม Open source ที่นิยมนำมาป็นเครื่องมือตรวจสอบแจ้งเตือนการถูกโจมตีทางเครือข่ายคอมพิวเตอร์ (Tongpagdee, 2561) ได้อธิบายว่า “การทำงานของโปรแกรม Snort จะใช้ไลบรารีพื้นฐานชื่อ libpcap ซึ่งนิยมใช้กันใน network sniffer และ network analyzer สำหรับโปรแกรม Snort ยังสามารถทำ protocol analysis, content searching หรือ matching, ตรวจสอบการถูกโจมตีและ probe เช่น buffer overflow, stealth port scan, CGI attack, SMB probe, OS Fingerprint และอื่นๆ” นอกจากนี้ยังมีงานวิจัยของ (Karahana & Berat, 2020) ได้ประยุกต์ใช้โปรแกรม Snort มาทำ IDS (Intrusion detection system) หรือระบบตรวจจับการถูกโจมตีทางเครือข่ายเพื่อรักษาความปลอดภัย ตรวจสอบความพยายามที่จะโจมตีเข้ามายังเครือข่าย โดยระบบจะแจ้งเตือนผู้ดูแลระบบเมื่อมีการโจมตีหรือมีการพยายามที่จะโจมตีเครือข่าย โดย IDS ไม่ใช่ระบบที่ใช้ป้องกันการถูกโจมตีแต่เป็นระบบที่แจ้งเตือนภัยเท่านั้น ปัญหาที่พบเจอของ IDS คือบางที่ IDS ไม่สามารถตรวจพบการโจมตีระบบเครือข่ายได้เนื่องจากมีปัญหาเรื่องการทำงานกับสถานะแวดล้อมของอุปกรณ์ Network และปัญหาที่สำคัญของ IDS คือ IDS ไม่สามารถป้องกันการถูกโจมตีได้โดยทันทีหรือแบบเรียลไทม์ เช่น การโจมตีแบบ DoS (Denial of Services) หรือ DDos (Distributed Denial of Services) จากปัญหาเหล่านี้จึงมีการคิดค้นระบบที่เรียกว่า IPS (Intrusion Prevention System) ซึ่งสามารถตรวจจับการถูกโจมตีและหยุดการโจมตีได้อย่างทันที ในปัจจุบันภัยคุกคามจากอินเทอร์เน็ตและระบบเครือข่ายคอมพิวเตอร์มีแนวโน้มสูงมากขึ้น การติดตั้งไฟร์วอลล์อย่างเดียวยังไม่เพียงพอIDS/IPS จึงเป็นสิ่งจำเป็นที่จะต้องมีการรักษาความปลอดภัยในเครือข่ายของทุกองค์กรต่อไปนี่คือเหตุผลว่าทำไมต้องมีระบบ IDS/IPS

1. เพื่อใช้ตรวจสอบสำหรับค้นหาต้นเหตุแหล่งที่มาที่โจมตีหรือบุกรุก
2. เพื่อตรวจจับความพยายามที่จะโจมตีเครือข่ายและป้องกันก่อนที่จะถูกโจมตี
3. เพื่อเก็บสถิติความพยายามที่โจมตี และนำข้อมูลไปวิเคราะห์ภัยคุกคามที่อาจจะเกิดขึ้นกับองค์กร

4. เพื่อเป็นเครื่องมือในการวัดประสิทธิภาพในการป้องกันภัยของระบบการรักษาความปลอดภัยอื่นเช่นไฟร์วอลล์ เป็นต้น

นอกจากนี้ (Gogoi, 2018) ได้ใช้งานโปรแกรม Snort ตรวจสอบการโจมตีแบบ DDoS ได้ทั้ง ICMP flooding และ SYN flooding และยังสามารถค้นหาไอพีของผู้ที่อยู่เบื้องหลังการโจมตีได้ และยังได้แสดงโครงสร้างของโปรแกรม Snort ดังภาพ 15



ภาพ 15 Snort Architecture (Gogoi, 2018)

ดังนั้นในการออกแบบระบบถ่ายโอนข้อมูลแบบกึ่งปิดจึงได้นำเอา Snort มาใช้งานเพื่อเพิ่มประสิทธิภาพด้านความปลอดภัยให้กับระบบร่วมกับไฟร์วอลล์

12. Firewall

(ThaiFirewall, 2564) ได้ระบุว่า Firewall เป็นระบบรักษาความปลอดภัยของเครื่องคอมพิวเตอร์ ไม่ให้ถูกโจมตีจากผู้ไม่หวังดีหรือการสื่อสารที่ไม่ได้รับอนุญาต ซึ่งส่วนใหญ่จะมาจากระบบเครือข่ายอินเทอร์เน็ต รวมถึงเครือข่าย LAN ในปัจจุบัน Firewall มีทั้งอุปกรณ์ที่เป็น Hardware และ Software

1. Software Firewall

เป็นโปรแกรม Firewall ที่ต้องมีการติดตั้งลงบนเครื่อง PC (Personal Server) หรือบนเครื่องแม่ข่าย ซึ่ง Software Firewall มีให้เลือกใช้งานทั้งแบบฟรีที่เป็น Open Source เช่น

IPTABLES , IPCOP , Endian Firewall และแบบมีการคิดค่า License เช่น Kerio WinRoute Firewall ,ISA Firewall

2. Hardware Firewall

เป็น Firewall ที่มีประสิทธิภาพการทำงานที่เร็วกว่า Software Firewall เนื่องจาก Hardware Firewall มีการประมวลผลโดยใช้ ASIC CHIP ซึ่งปัจจุบันมีการรวมความสามารถหลาย ๆ อย่างเข้ามาในอุปกรณ์ประเภท Hardware Firewall เช่น Anti-Virus, Anti-Spam , IPS

Firewall ทำหน้าที่ตรวจสอบการเชื่อมต่อต่าง ๆ ให้เป็นไปตามกฎ ซึ่ง Firewall จะเป็นตัวกรองข้อมูลและวิเคราะห์ว่าข้อมูลชนิดนี้คืออะไร (Source) ตัวข้อมูลต้องการจะไปที่ไหน (Destination) และข้อมูลชิ้นนี้จะบริการอะไรหรือทำอะไร (Service/Port) ถ้ารู้ว่าข้อมูลไม่ปลอดภัย หรือมีความเสี่ยงที่จะมาทำความเสียหาย Firewall ก็จะทำหน้าที่กั้นไม่ให้ข้อมูลเข้าไปได้ ประเภทของ Firewall มี 5 ประเภทดังนี้

1. Packet Filtering Firewall จะทำหน้าที่พิจารณาเปรียบเทียบ Packet กับกฎที่ผู้ใช้งาน กำหนดไว้ถ้าผ่านเชื่อถือจะส่ง Packet ไปยังปลายทางแต่ถ้าไม่น่าเชื่อถือจะปฏิเสธ
2. Circuit-level Gateway ทำหน้าที่ตรวจสอบเส้นทางการเชื่อมต่อเครือข่าย และจะสร้างเส้นทางเสมือนขึ้นมาเพื่อพิจารณาว่าเครือข่ายที่เข้ามามีความน่าเชื่อถือหรือไม่ Firewall ประเภทนี้จะไม่สามารถตรวจสอบ Packet เองได้ แต่การตรวจสอบ Packet ของ Firewall จะทำงานบน Transport Layer ใน OSI Model
3. Stateful Inspection Firewall ทำหน้าที่ตรวจสอบสถานะ ตรวจสอบ Packet และยังติดตาม Packet นั้นว่าเคยเข้ามาในเครือข่ายนี้แล้ว หรือเคยเข้ามาครั้งแรก โดยจะนำเอาข้อมูลของ Packet และข้อมูลที่ได้จาก Packet ก่อนหน้านั้นมาพิจารณารวมกัน ซึ่งประเภทนี้ จะมีความปลอดภัยมากกว่าการตรวจสอบเส้นทาง หรือการกรอง Packet เพียงอย่างเดียว
4. Application-level Gateway เป็น Firewall ชนิดที่ติดตั้งบนเครื่องคอมพิวเตอร์แยกตัวออกจากเครื่อง Router แต่ยังคงเชื่อมต่อกับเครื่อง Router เพื่อค้นหาเส้นทางของการส่ง Packet ทำหน้าที่กรอง และตรวจสอบดูแลเนื้อหาของ Packet สามารถตรวจจับ และปิดกั้นการโจมตีที่มองไม่เห็นบนเครือข่าย OSI Model ได้ บางครั้งทำหน้าที่คล้าย Proxy Firewall ที่เป็นระบบรักษาความปลอดภัยเครือข่ายที่ปกป้องข้อมูลเครือข่ายโดยการควบคุม และตรวจสอบข้อมูลที่มีความผิดปกติได้
5. Next-generation Firewall ทำหน้าที่รวมการตรวจสอบเส้นทางเครือข่ายเข้ากับการตรวจสอบ Packet และยังรวมถึง Deep Packet Inspection (DPI) ซึ่งเป็นวิธีการขั้นสูงในการตรวจสอบ และจัดการรับส่งข้อมูลเครือข่าย ถือเป็นการรวมรูปแบบของ Packet ที่

หลากหลาย รวมทั้งระบบรักษาความปลอดภัยเครือข่ายอื่น ๆ เช่น การตรวจจับหรือป้องกันการบุกรุกการกรองมัลแวร์ และโปรแกรมป้องกันไวรัส

ก่อนการเลือกใช้งาน Firewall จะต้องเข้าใจก่อนว่าแต่ละประเภทของ Firewall ทำงานแบบใด มีข้อแตกต่างกันอย่างไร รวมทั้งต้องวิเคราะห์ความต้องการว่าเราจะนำ Firewall มาใช้ในด้านใดบ้าง

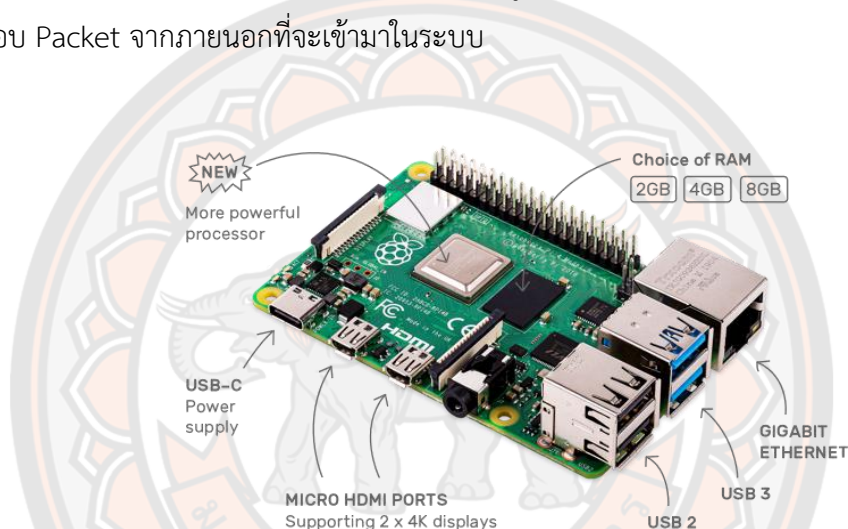
ตาราง 1 เปรียบเทียบข้อดีข้อเสียของ Firewall แต่ละประเภท (kankann, 2563)

ประเภท	ข้อดี	ข้อเสีย
Packet Filtering Firewall	มีประสิทธิภาพในการประมวลผล Packet	มีความเสี่ยงในการถูกโจมตี
Circuit-level Gateway	การรับส่งข้อมูลและการประมวลผลมีประสิทธิภาพสูงกว่าระดับ Application-level Gateway	ไม่สามารถกรองเนื้อหาของข้อมูลที่จะเข้ามาได้
Stateful Inspection Firewall	สามารถปิดกั้นและป้องกันการโจมตีที่ช่องโหว่ Protocol ได้	ต้องใช้ทักษะระดับสูงในการกำหนดค่าเพื่อความปลอดภัย
Application-level Gateway	มีความสามารถในการตรวจจับและปิดกั้นการโจมตีที่มองไม่เห็นบนเครือข่ายแบบจำลอง OSI	มีค่าใช้จ่ายในการประมวลผลสูงและต้องมีการตั้งค่า Proxy สำหรับแอปพลิเคชันเครือข่ายทุกตัวที่ใช้งานอยู่
Next-generation Firewall	รวมความสามารถของ Firewall ประเภทอื่นๆ และรวมความสามารถในเรื่องความปลอดภัย รวมถึงระบบตรวจจับ / ป้องกันการบุกรุก, ภัยคุกคามขั้นสูง และการสแกนมัลแวร์	ต้องใช้งบลงทุนสูงทั้งในการกำหนดกฎ และปรับปรุงให้ Firewall สามารถทำงานบนเครือข่ายที่มีความซับซ้อนได้อย่างมีประสิทธิภาพ

ในการออกแบบระบบถ่ายโอนข้อมูลแบบกึ่งปิดจะนำ Firewall ที่เป็น Software firewall ทำหน้าที่ตรวจสอบ Packet จากภายนอกที่จะส่งเข้ามาในระบบและให้ Firewall ควบคุมการทำงานของ port การเชื่อมต่อให้ packet ออกได้อย่างเดียว

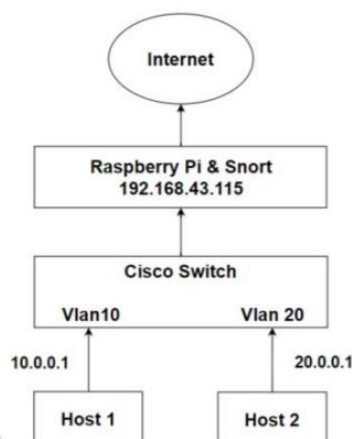
13. Raspberry Pi

(PoundXI, 2560) ได้กล่าวว่ Raspberry Pi เป็นบอร์ดคอมพิวเตอร์ขนาดเล็ก (Single-Board Computer หรือ SBC) ที่ถูกพัฒนาขึ้นโดย Raspberry Pi Foundation มีคุณสมบัติเด่น คือ ติดต่อสื่อสาร และเป็นคอมพิวเตอร์ที่รวบรวมอัลกอริทึมของ Internet of things (IoT) หรือจะใช้ทำงานด้านควบคุมหุ่นยนต์และอุปกรณ์อิเล็กทรอนิกส์ Raspberry Pi ใช้ลินุกซ์เป็นระบบปฏิบัติการ โดยระบบปฏิบัติการที่นิยมใช้กัน คือ ระบบปฏิบัติการ Raspbian เพราะเป็นระบบปฏิบัติการที่ถูสนับสนุนโดยตรงจากทาง Raspberry Pi Foundation ซึ่งในงานวิจัยนี้จะใช้งาน Raspberry Pi4 ดังภาพ 16 เป็นตัวจัดการเปิดปิดอินเทอร์เน็ตเฟสของโมดูล Network และใช้ทำหน้าที่เป็น Firewall ตรวจสอบ Packet จากภายนอกที่จะเข้ามาในระบบ



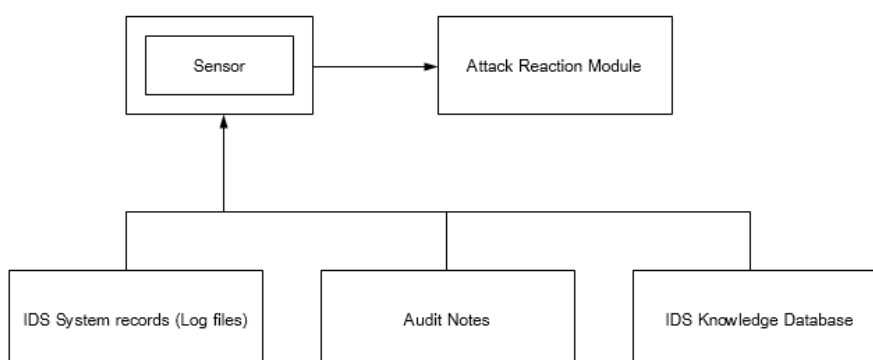
ภาพ 16 Raspberry Pi4 model B (Raspberrypi.org, 2021)

นอกจากนี้ยังมีงานวิจัยที่ได้นำเอา Raspberry Pi มาทำหน้าที่เป็น Firewall สำหรับธุรกิจขนาดเล็กถึงขนาดกลาง (Karahana & Berat, 2020) ได้นำเสนอบทความวิจัยเรื่อง Raspberry Pi Firewall and Intrusion Detection System ซึ่งได้นำเสนอวิธีการกำหนดค่า Firewall ในระดับพื้นฐานและขั้นสูงสำหรับธุรกิจขนาดเล็กไปจนถึงขนาดกลางโดยการนำเอา Raspberry Pi มาติดตั้ง Firewall และ Snort ไว้ก่อน Network Switch เพื่อป้องกันผู้บุกรุกดังภาพสถาปัตยกรรมระบบดังภาพ 17



ภาพ 17 Designed system architecture (Karahana & Berat, 2020)

และยังมี Intrusion Detection Systems ดังภาพที่ 18 เป็นซอฟต์แวร์สำหรับการตรวจสอบกิจกรรมที่เป็นอันตรายหรือการละเมิดนโยบายกับเครือข่ายหรือระบบ ระบบ IDS มีฐานข้อมูลที่ตรวจจับภัยคุกคามที่เกิดขึ้นบ่อยกับเครือข่ายมีการบันทึกเหตุการณ์ที่เกี่ยวข้องหยุดการโจมตีและรายงานต่อผู้ดูแลระบบความปลอดภัย เมื่อเกิดการโจมตีนอกจากนี้ยังกำหนดค่าอุปกรณ์เครือข่ายใหม่เช่น Firewall หรือเราเตอร์บล็อกการโจมตีในลักษณะเดียวกัน



ภาพ 18 Intrusion Detection Systems (Karahana & Berat, 2020)

จากการศึกษาสรุปได้ว่ามีแอปพลิเคชันและอุปกรณ์ในตลาดมีหลากหลายชนิดที่มีฟีเจอร์สำหรับตรวจจับการบุกรุกจึงควรเลือกใช้งานให้เหมาะสมกับความต้องการ การศึกษานี้ได้แสดงระบบ Proxy ตรวจจับการบุกรุกที่ประหยัด โดยใช้ Raspberry Pi และโปรแกรม Open source Snort โดย

ได้ทำการทดสอบด้วยการโจมตีพื้นฐานสองแบบเพื่อดูว่าระบบสามารถตรวจจับเหตุการณ์ที่ไม่ต้องการได้หรือไม่บนเครือข่ายที่ถูกขยายด้วย Network Switch และผลลัพธ์แสดงให้เห็นว่าระบบทำงานได้สำเร็จในเครือข่ายขนาดเล็กและในอนาคตสามารถทดสอบระบบกับจำนวนโฮสต์ที่มากขึ้นและการโจมตีที่ซับซ้อนมากขึ้น งานวิจัยนี้เกี่ยวข้องกับงานที่ทำตรงที่ใช้ Raspberry Pi ติดตั้ง Firewall และ Snort เป็นตัวกลางป้องกันการโจมตีจาก Network ภายนอกที่จะโจมตีเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิทยาศาสตร์ และแตกต่างกันตรงที่งานในวิจัยนี้ได้นำเสนอการป้องกันและแจ้งเตือนการโจมตีเพียงอย่างเดียวส่วนงานที่ทำอยู่จะมีการปกป้องข้อมูลด้วยการเข้ารหัสข้อมูล นอกจากนี้ก็มีผลงานวิจัยที่ได้นำเอา Raspberry Pi firewall สำหรับหน่วยงานธุรกิจขนาดเล็ก (SMEs) (นพดล จินตสุนทรอุไร ตรีรัตน์ เมตต์การุณจิต, 2558) ได้นำเสนอแนวคิดเกี่ยวกับการนำบอร์ด Raspberry Pi ที่เป็นบอร์ดสมองกลฝังตัว (Embedded board) ชนิดหนึ่งที่เป็นที่นิยมในปัจจุบันประยุกต์การใช้งานเป็น firewall ที่ได้เปรียบเทียบกับ Raspberry Pi firewall กับบอร์ดสมองกลอื่นๆ ที่ใช้เป็น firewall และ firewall ของเราเตอร์ขนาดกลาง ซึ่งผลออกมานั้นเห็นได้ว่า Raspberry Pi นั้นเมื่อเทียบกับคิวบ์บอร์ดประสิทธิภาพการทำงานในแง่ของทราฟฟิค (throughput) กับจำนวนกฎของ firewall คิวบ์บอร์ดทำงานได้ดีกว่า Raspberry Pi firewall แต่ก็ไม่แตกต่างกันมาก นอกจากนี้ Raspberry Pi firewall มีความสามารถเทียบเคียงกับ firewall ของเราเตอร์ขนาดกลางในแง่การตั้งกฎของ firewall รวมถึงการตรวจจับและป้องกันภัยคุกคาม โดยจุดเด่นของ Raspberry Pi firewall คือมีราคาถูกกว่าเราเตอร์ขนาดกลางมาก ที่เป็นทางเลือกสำหรับธุรกิจขนาดกลางและขนาดเล็กที่จะนำไปใช้งานได้

14. Tkinter

Tkinter หรือ TK Interface (digitalschool.club, 2565) เป็น library ของภาษา python ที่ใช้พัฒนา GUI ของภาษา python ซึ่ง TK จะประกอบไปด้วย 4 องค์ประกอบหลัก ๆ ที่สำคัญคือ widgets, geometry management, event handling และ Command Callbacks มีรายละเอียดดังนี้

1. Widgets คือองค์ประกอบต่างๆที่ปรากฏบนหน้าจอ เช่น Button, Label, Frame, checkbox, tree views, scrollbars, text areas เป็นต้น Widgets ต่าง ๆ เหล่านี้ถูกออกแบบให้มีลักษณะการทำงานแบบลำดับชั้นดังนั้นการสร้าง GUI ใด ๆ ให้ปรากฏบนจอภาพ จะเริ่มต้นสร้างจากลำดับชั้น ที่เรียกว่า root window (root) เป็นอันดับที่ 1 ลำดับชั้นที่ 2 จึงสร้างเฟรม (Content frame) เพื่อบรรจุ widgets ต่างๆ ลงบน root window และในลำดับชั้นที่ 3 เป็นการเพิ่ม Widgets ต่าง ๆ ที่ได้ออกแบบไว้ลงบนเฟรม ที่ได้จัดเตรียม

ไว้ สำหรับการควบคุมการทำงานของ Widgets จะทำงานเป็นแบบลำดับชั้น Widgets ใน ภาษาไพธอนถูกเขียนขึ้นด้วยโปรแกรมเชิงวัตถุ ดังนั้นทุกๆ Widget ที่สร้างขึ้นจะถูกเรียกว่า อ็อบเจกต์ (Object) หรือวัตถุ เมื่อทำการสร้างอินสแตนซ์ของ Widget ใดๆ ขึ้นจะต้องส่ง พารามิเตอร์ให้กับคลาสแม่ตามลำดับชั้น ยกเว้น Root ซึ่งเป็นคลาสแม่ที่อยู่ในตำแหน่งบนสุด (Top level window) ของลำดับชั้น คลาสลูกทุก ๆ คลาสจะถูกสร้างภายใน root เท่านั้น การเรียกใช้งาน Widget แสดงดังภาพ 19 ซึ่งเป็นตัวอย่างการสร้าง Root, Content frame และ Widgets

```
from tkinter import *
root = Tk() # Create root window
content = Frame(root) # Create content frame
button = Button(content) # Create button in frame
```

ภาพ 19 ตัวอย่างการสร้าง Root, Content frame และ Widgets

2. Geometry management คือการจัดการรูปร่างเรขาคณิตให้กับ Widgets การวาง Widgets ลงบนเฟรมนั้นจะต้องกำหนดตำแหน่งในการวาง โดยอาศัยศาสตร์ทางด้าน เรขาคณิตเข้าช่วย เพื่อให้ Widgets ที่จะวางอยู่ในตำแหน่งที่เหมาะสม ซึ่งไพธอนมี 3 เมธอด ในการจัดการเกี่ยวกับเรขาคณิตของ Widgets ประกอบไปด้วยเมธอด pack(), grid() และ place() สำหรับปัญหาการวาง Widgets ที่เกิดขึ้นเสมอ คือ การปรับขนาดของเฟรมหรือ Widgets จะส่งผลกระทบต่อซึ่งกันและกัน เช่น ถ้าผู้ใช้งานย่อหรือขยายขนาดของหน้าต่างหลัก จะส่งผลกระทบต่ออ็อบเจกต์ต่าง ๆ ที่อยู่ภายในหน้าต่างหลักนั้น ๆ ทั้งนี้ ซึ่งอาจจะทำให้ปุ่ม ตัวอักษร ลาเบล เกิดความผิดเพี้ยนไปจากเดิม ปัญหาต่าง ๆ เหล่านี้จะถูกจัดการด้วย Geometry management ที่อยู่ใน Tk โดยใช้เทคนิคที่เรียกว่า Master and Slave โดย Master คืออ็อบเจกต์ที่ทำหน้าที่รองรับ Widgets ต่าง ๆ ที่จะทำงาน เช่น root หรือ content frame สำหรับ Slave คือ Widgets ต่าง ๆ ที่วาดหรือวางลงบน Master สำหรับ การทำงานของ Geometry management นั้นจะใช้วิธีสอบถามไปยัง Widgets ต่าง ๆ ที่ กำลังจะทำงานว่าแต่ละ Widgets ต้องการพื้นที่ ๆ ใช้สำหรับการทำมายน้อยเพียงใด จากนั้น Geometry management จะคำนวณพื้นที่ทั้งหมดในภาพรวม เพื่อจัดวาง Widgets เหล่านั้นในตำแหน่งที่เหมาะสมต่อไป
3. Event handling คือ การจัดการกับเหตุการณ์ต่าง ๆ ที่ผู้ใช้งานกระทำกับ Widgets ใด ๆ บน GUI เช่น การกดปุ่ม การกดปุ่มใด ๆ บนแป้นพิมพ์ การเคลื่อนเมาส์ การปรับขนาดของ

หน้าต่างวินโดวส์ เป็นต้น ซึ่งเหตุการณ์ต่าง ๆ เหล่านี้จะถูกจัดการโดย Tk ซึ่งเรียกว่า event loop โดยจะทำงานร่วมกับระบบปฏิบัติการโดยตรง เช่น เมื่อเคอร์เซอร์เมาส์ไปยังปุ่มจะส่งผลให้ปุ่มดังกล่าวจะเปลี่ยนสี และเมื่อเคอร์เซอร์ออกจากปุ่มจะทำให้สีของปุ่มกลับไปเป็นสีเดิม เป็นต้น

4. Command Callbacks คือการตอบสนองต่อเหตุการณ์ที่เกิดขึ้น มีหลาย Widgets จำเป็นต้องกระทำอย่างใดอย่างหนึ่งเมื่อมีการคลิกหรือกระทำกับ Widgets เหล่านั้น ในไพธอนจะใช้คำสั่ง "command" หรือเรียกว่า "Callbacks" ในการตอบสนองต่อเหตุการณ์ต่าง ๆ ที่เกิดขึ้นกับ Widgets โดยมีรูปแบบคำสั่งคือ `command = functionName`



บทที่ 3

วิธีดำเนินงานวิจัย

ในบทนี้จะกล่าวถึงวิธีการดำเนินการทำวิจัยโดยจะแบ่งออกเป็น 5 ขั้นตอนได้แก่การกำหนดปัญหา, ศึกษาเอกสารและงานวิจัยที่เกี่ยวข้อง, วิเคราะห์และออกแบบสถาปัตยกรรมระบบ, ขั้นตอนการติดตั้งและการทำงานของระบบ, วิธีการทดลองประเมินผล มีรายละเอียดดังนี้

1. กำหนดปัญหา

ได้ศึกษาและวิเคราะห์ปัญหาจากศูนย์เครื่องมือคณะวิทยาศาสตร์ซึ่งมีรายละเอียดดังนี้

- 1.1. การนำข้อมูลผลการวิเคราะห์ตัวอย่างออกจากเครื่องคอมพิวเตอร์ควบคุมการทำงานเกิดความไม่สะดวกกับนักวิทยาศาสตร์นักวิจัยและผู้รับบริการเนื่องจากต้องเขียนลงแผ่น CD หรือ DVD เท่านั้น
- 1.2. ป้องกันเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิเคราะห์ติด Malware และไวรัสคอมพิวเตอร์จากข้อที่ 1.1 และจากวิธีการอื่น ๆ ซึ่งมีผลสามารถทำให้เกิดปัญหาติด Malware และ ไวรัสคอมพิวเตอร์ถ้ามีการนำอุปกรณ์ไปต่อหรือเชื่อมต่อกับเครือข่าย Internet
- 1.3. จำเป็นที่จะต้องพัฒนาเครื่องมือที่สามารถแก้ปัญหาในข้อ 1.1 และข้อ 1.2 เพื่อเพิ่มความสะดวกในการนำข้อมูลผลการวิเคราะห์ออกมาใช้งานอย่างปลอดภัย

2. ศึกษาเอกสารและงานวิจัยที่เกี่ยวข้อง

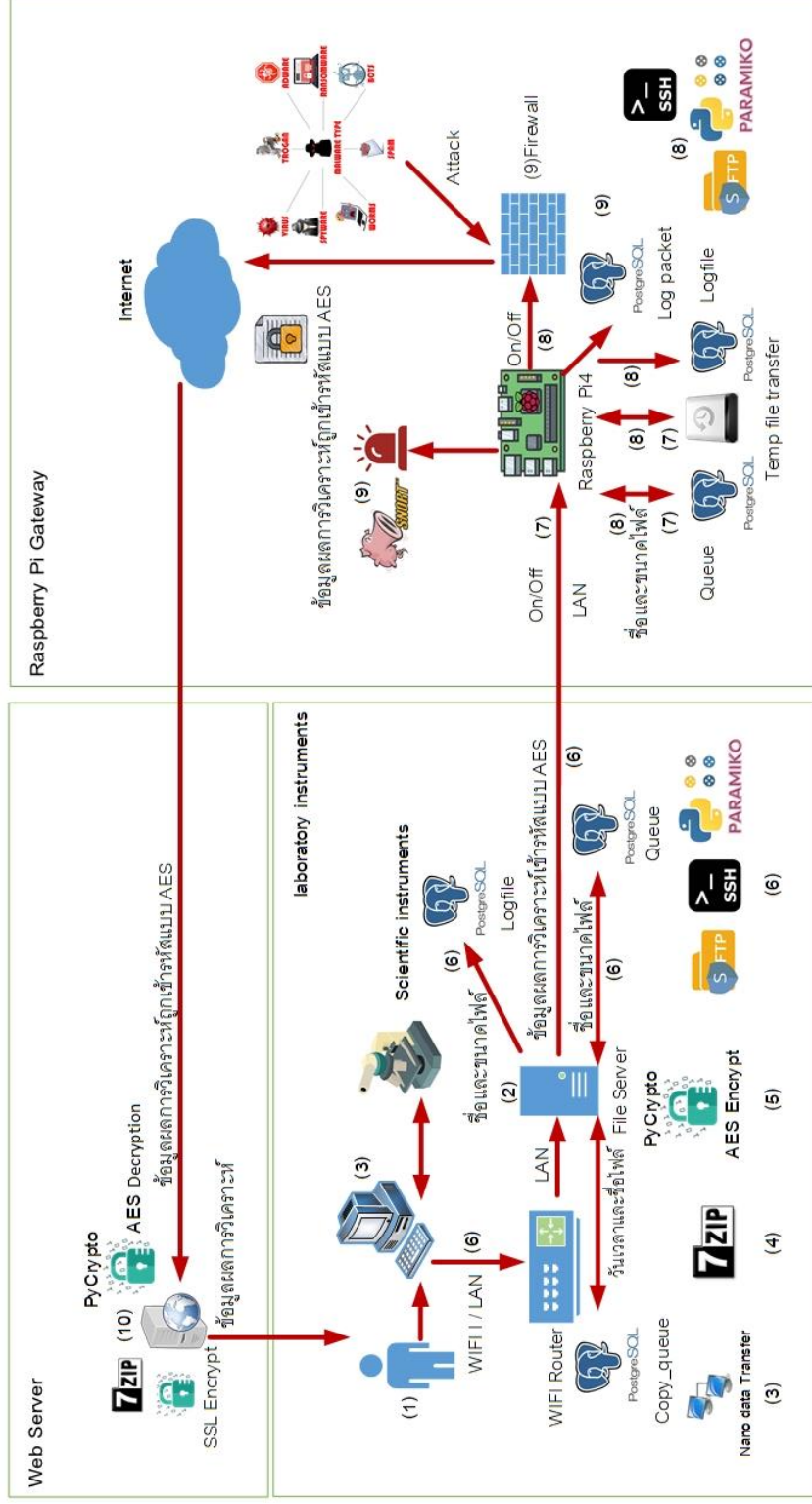
- 2.1. ศึกษาหลักการ CPS เพื่อออกแบบโครงสร้างระบบเพื่อแก้ไขปัญหาด้านความปลอดภัยในการโอนถ่ายข้อมูลจากเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิทยาศาสตร์จากภัยคุกคามภายนอกเช่นการโจมตีจาก malware และไวรัสคอมพิวเตอร์เข้ามาทำความเสียหายให้กับเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิทยาศาสตร์และเพื่อเพิ่มความสะดวกในการนำออกไฟล์ผลการวิเคราะห์ให้กับนักวิจัยด้วยความรวดเร็วและเป็นปัจจุบัน โดยหลักการ CPS เป็นการนำเอาระบบทางกายภาพเชื่อมต่อเข้ากับระบบไซเบอร์ ดังนั้นจึงได้ออกแบบระบบโดยการนำเอาเครื่องคอมพิวเตอร์ที่ควบคุมการทำงานของเครื่องมือวิเคราะห์ทางวิทยาศาสตร์ในส่วนของกายภาพเชื่อมต่อเข้ากับระบบไซเบอร์คือไฟล์เซิร์ฟเวอร์เพื่อทำการประมวลผลและจัดการข้อมูลผลการวิเคราะห์ผ่านตัวกลางอย่าง

Raspberry Pi4 ที่เชื่อมต่อกับเครือข่ายอินเทอร์เน็ตไปยังเครื่องให้บริการข้อมูลเช่น เว็บเซิร์ฟเวอร์

- 2.2. ศึกษาการใช้งาน Raspberry Pi4 เพื่อเป็นตัวกลางโอนถ่ายข้อมูลและตรวจสอบป้องกันการโจมตี โดยนำ Raspberry Pi4 ทำหน้าที่เป็น gateway ซึ่งการทำงานจะประกอบไปด้วย การเปิด-ปิด Network interface, Firewall, Snort, log และการแจ้งเตือนเมื่อมีการโจมตีจากภายนอก
- 2.3. ศึกษา Paramiko library ใช้โปรโตคอลสำหรับโอนถ่ายข้อมูล sftp โอนถ่ายข้อมูลจากเครื่องมือควบคุมการทำงานของเครื่องมือวิทยาศาสตร์ไปยัง File server จาก File server ไปยัง Raspberry Pi4 และจาก Raspberry Pi4 ไปยังเครื่องบริการข้อมูล Web server
- 2.4. ศึกษาการใช้ Firewall ในการควบคุมการทำงานของ Port การเชื่อมต่อและกรองตรวจสอบ Packet ที่จะเข้าและออกจากระบบโดยจะติดตั้ง Firewall ใน Raspberry Pi4 ซึ่งเป็นส่วนแรกที่ติดต่อกับเครือข่ายอินเทอร์เน็ต
- 2.5. ศึกษาการใช้งานโปรแกรม Snort ใช้เป็นเครื่องมือตรวจจับการบุกรุกทางเครือข่ายโดยจะติดตั้งใน Raspberry Pi4 ซึ่งเป็นส่วนแรกที่ติดต่อกับเครือข่ายอินเทอร์เน็ต
- 2.6. ศึกษาการใช้งานโปรแกรม 7zip ใช้ในการบีบอัดและแบ่งไฟล์ผลการวิเคราะห์ให้เป็นไฟล์ขนาดเล็กลงเพื่อเพิ่มความเร็วในการโอนถ่ายข้อมูล โดยติดตั้งที่ File server
- 2.7. ศึกษาการเข้ารหัสและถอดรหัสลับ AES เพื่อใช้เข้ารหัสและถอดรหัสข้อมูลผลการวิเคราะห์เพื่อเพิ่มความปลอดภัยของข้อมูลที่อาจจะถูกผู้ไม่หวังดีดักจับข้อมูลระหว่างการโอนถ่ายโดยใช้ PyCrypto ซึ่งเป็น library ของ python รองรับเข้ารหัสและถอดรหัส เช่น SHA256 ,RIPEMD160 ,AES, DES, RSA, ElGamal, และอื่น ๆ เป็นต้น รองรับทั้ง Python 2 และ Python 3 โดยจะติดตั้งที่ File server เพื่อทำการเข้ารหัส และ Web server เพื่อทำการถอดรหัส
- 2.8. ศึกษาการใช้งาน Tkinter library เพื่อใช้สร้างโปรแกรมถ่ายโอนข้อมูลผลการวิเคราะห์ร่วมกับ Paramiko library ติดตั้งไว้ที่เครื่องควบคุมเครื่องมือวิเคราะห์เพื่อเป็นเครื่องมือช่วยนำผลการวิเคราะห์ไปยัง File server เพื่อทำการบีบอัดข้อมูลผลการวิเคราะห์ เข้ารหัสลับไฟล์ที่ถูกบีบอัดแบบ AES
- 2.9. ศึกษาภาษา Python, PHP, Postgresql และฐานข้อมูล MySQL ใช้จัดการข้อมูลผลการวิเคราะห์โดยใช้ภาษา PHP พัฒนาเว็บแอปพลิเคชัน และใช้ฐานข้อมูล MySQL เป็น database จัดเก็บข้อมูลผลการวิเคราะห์
- 2.10. ทำการศึกษางานวิจัยที่เกี่ยวข้องเพื่อนำมาประยุกต์ใช้ในงานวิจัย

3. วิเคราะห์และออกแบบสถาปัตยกรรมระบบ

Semi closed data transfer system



ภาพ 20 สถาปัตยกรรมของระบบ

จากภาพ 20 สามารถอธิบายการทำงานได้ดังนี้

1. นักวิทยาศาสตร์ นักวิจัย ทำการวิเคราะห์ตัวอย่าง
2. นักวิทยาศาสตร์ นักวิจัย นำผลการวิเคราะห์มาเก็บไว้ใน File server สำหรับเก็บผลการวิเคราะห์ด้วยเครื่องมือที่พัฒนาขึ้นชื่อ Nano data transfer โดยจะเก็บข้อมูลผลการวิเคราะห์ไว้ในโฟลเดอร์ File_book และสร้าง copy queue ในฐานข้อมูลเพื่อเก็บลำดับคิวข้อมูลผลการวิเคราะห์
3. ระบบทำการตรวจสอบข้อมูลในฐานข้อมูลว่ามีคิวข้อมูลผลการวิเคราะห์หรือไม่
4. ถ้ามีคิวข้อมูลผลการวิเคราะห์ให้ตรวจสอบไฟล์หรือโฟลเดอร์ใน File_book ของ file server และทำการบีบอัดไฟล์หรือโฟลเดอร์ ถ้าไฟล์หรือโฟลเดอร์มีขนาดใหญ่ให้ทำการแบ่งไฟล์ที่บีบอัดให้เป็นไฟล์ขนาดเล็กๆด้วยโปรแกรม 7zip ตามลำดับคิวที่อ่านได้จากฐานข้อมูล
5. ทำการเข้ารหัสลับไฟล์ที่ถูกบีบอัดด้วยโปรแกรม 7zip แบบ AES ด้วย PyCrypto
6. หาขนาดไฟล์ นำชื่อไฟล์และขนาดไฟล์มาเก็บไว้ในฐานข้อมูลเพื่อสร้าง queue, logfile และเก็บไฟล์ที่ถูกบีบอัดพร้อมเข้ารหัสลับไว้ในโฟลเดอร์ Temp_file ของ File server ทำการลบคิวใน copy queue ที่ทำเสร็จ
7. เริ่มการโอนถ่ายข้อมูล File server ตรวจสอบ connection ของ Raspberry Pi4 ถ้ายังไม่มี connection ก็ให้ตรวจสอบต่อไปเรื่อยๆ แต่ถ้ามี connection ให้ File server เชื่อมต่อกับ Raspberry Pi4 และอีกอินเตอร์เฟซของ Network ของ Raspberry Pi4 จะปิดไม่มีการเชื่อมต่อ Network ภายนอก เริ่มการโอนถ่ายข้อมูลตามคิวโดยใช้ โปรโตคอล SFTP ของ Paramiko library ไปยัง Raspberry Pi4 ในระหว่างโอนถ่ายข้อมูลจะมีการตรวจสอบความสมบูรณ์ของไฟล์ไฟล์ข้อมูลไหนโอนถ่ายเสร็จสมบูรณ์แล้วให้ลบรายการชื่อไฟล์นั้นออกจาก Queue และลบไฟล์นั้นจากโฟลเดอร์ Temp_file ของ File server เมื่อโอนไฟล์ถึงเวลาที่กำหนดระบบจะปิดการเชื่อมต่อระหว่าง File server กับ Raspberry Pi4 และเปิด connection กับ Web server
8. Raspberry Pi4 จะทำหน้าที่เปิดปิด connection ตามช่วงเวลาที่กำหนด ถ้ามี connection กับ File server และมีไฟล์ข้อมูลส่งมาให้ทำการสำรองข้อมูลใน Temp file transfer ไฟล์ไหนโอนถ่ายสมบูรณ์ให้สร้าง Queue ในฐานข้อมูลตามลำดับจนถึงเวลาที่กำหนดระบบจะปิด connection กับ File server
9. Raspberry Pi4 เชื่อมต่อ Wi-Fi / LAN เริ่มโอนถ่ายไฟล์ข้อมูลตามคิวโดยใช้ โปรโตคอล SFTP ของ Paramiko library ไป Web server มีการตรวจสอบความสมบูรณ์ของขนาดไฟล์ของแต่ละไฟล์ เมื่อไฟล์ไหนโอนถ่ายสมบูรณ์ให้ลบชื่อไฟล์รายการนั้นออกจาก Queue.พร้อมกับสร้างคิวใน Web server เขียน Logfile. เพื่อเก็บ log และทำการลบไฟล์นั้นออกจาก Temp file transfer

และเมื่อโอนถ่ายเสร็จสมบูรณ์ connection จะถูกปิดตัดการเชื่อมต่อกับ Web server และเปิด connection กับ File server

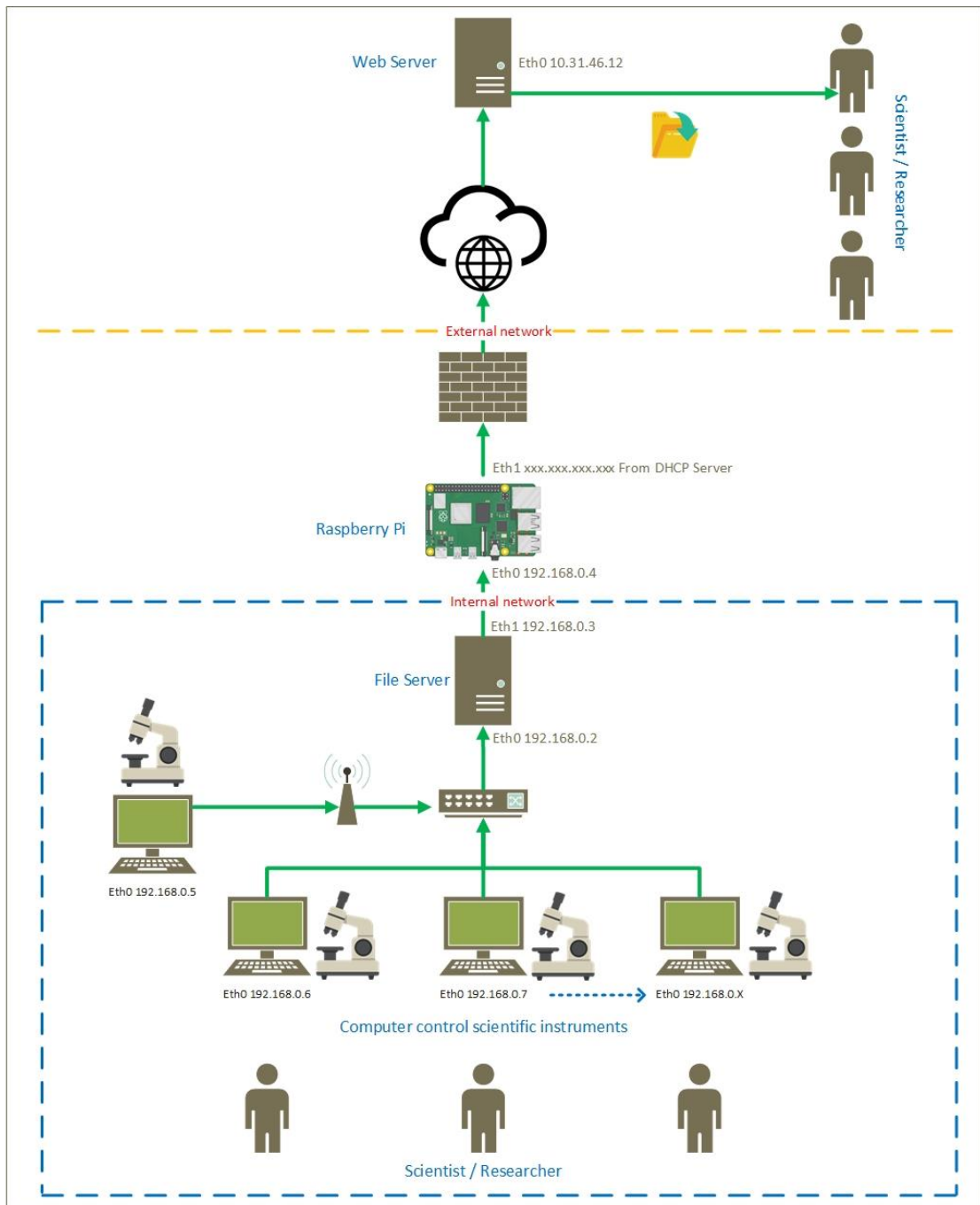
10. ในระหว่างโอนถ่ายข้อมูลไป web server ให้เก็บ Log packet ผ่าน Firewall และให้ Snort แจ้งเตือนเมื่อมี packets จากภายนอกพยายามผ่าน Firewall เข้ามาในระบบ
11. Web server ถอดรหัสและจัดการข้อมูลผลการวิเคราะห์และให้บริการผ่านทางเว็บไซต์

4. ขั้นตอนการติดตั้งและการทำงานของระบบ

ขั้นตอนการติดตั้งระบบและทดลองจะประกอบไปด้วยอุปกรณ์ดังนี้

1. คอมพิวเตอร์ 2 เครื่อง เพื่อใช้งานเป็น File server และเครื่องให้บริการไฟล์ผลการวิเคราะห์ Web server
2. Raspberry Pi4 1 เครื่องทำหน้าที่เป็นตัวกลางในการถ่ายโอนข้อมูลผลการวิเคราะห์จาก File server ไปยัง Web server และยังทำหน้าที่เป็น Firewall แจ้งเตือนการบุกรุกจากภายนอกที่จะเข้ามาในระบบ
3. USB hard drive ขนาด 32 Gb ติดตั้งไว้ใน Raspberry Pi4 เอาไว้เก็บ Temp file transfer
4. Network switch และ Wireless access point

หลังจากเตรียมอุปกรณ์เสร็จจะนำเอาอุปกรณ์มาต่อดังภาพ 21 โดยจะแบ่งโซนการทำงาน ออกเป็น Internal network และ External network โดย Internal network จะเป็นกลุ่มของเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมีวิทยศาสตร์และ File server ส่วน External network จะเป็นเครื่องให้บริการไฟล์ผลการวิเคราะห์ Web server มีตัวกลางชั้นเป็น Raspberry Pi4 Internal network จะไม่มีการเชื่อมต่อกับเครือข่ายภายนอกโดยตรง มีการกำหนด IP Address ให้กับอินเทอร์เน็ตเฟส Network ของอุปกรณ์ต่าง ๆ แยกออกจากเครือข่าย Network หลักเพื่อเพิ่มความปลอดภัย ส่วนที่มีการเชื่อมต่อเข้ากับเครือข่าย Network หลักคือ อินเทอร์เน็ตเฟส Network อันที่สองของ Raspberry Pi4 ที่รับ IP Address แบบอัตโนมัติจาก DHCP server สามารถเชื่อมต่อกับ Web server ได้

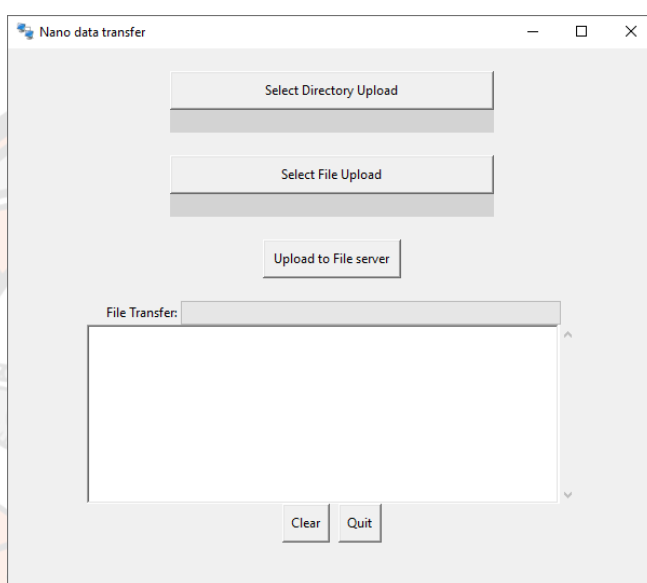


ภาพ 21 ขั้นตอนการติดตั้งระบบถ่ายโอนข้อมูลแบบกึ่งปิด

จากภาพ 21 อธิบายขั้นตอนการติดตั้งระบบถ่ายโอนข้อมูลแบบกึ่งปิดได้ดังนี้

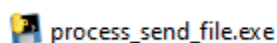
1. ส่วนของเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือจะถูกนำมาเชื่อมต่อเข้ากับ Network switch หรือ Wireless access point อย่างไม่อย่างหนึ่งขึ้นอยู่กับตำแหน่งที่อยู่ของเครื่องคอมพิวเตอร์เองว่าอยู่ใกล้กับ Network switch มากน้อยแค่ไหน ถ้าอยู่ใกล้กับ

Network switch จะใช้สาย Lan ในการเชื่อมต่อ แต่ถ้าอยู่ไกลจะใช้เป็น USB WIFI ในการเชื่อมต่อเพื่อแยก Network ออกมาเป็น Private network และมีการกำหนด IP Address แบบ Static โดยจะเริ่มจาก IP 192.168.0.5 ไปจนถึง IP 192.168.0.254 และเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือจะติดตั้งโปรแกรม Nano data transfer ดังภาพ 22 ที่พัฒนาใช้เป็นเครื่องมือสำหรับนักวิทยาศาสตร์หรือนักวิจัยที่มาใช้งานเครื่องมือโอนถ่ายข้อมูลผลการวิเคราะห์ไปยัง File server โดยใช้โปรโตคอล SFTP



ภาพ 22 โปรแกรม Nano data transfer

- ส่วนของ File Server จะมี Network interface card 2 อันคือ Eth0 และ Eth1 โดย Network interface card Eth0 จะเชื่อมต่อเข้ากับ Network switch กำหนด IP Address เป็น 192.168.0.2 และ Eth1 จะเชื่อมต่อกับ Network interface card Eth0 ของ Raspberry Pi4 ซึ่งเป็นตัวกลางในการถ่ายโอนข้อมูลผลการวิเคราะห์กำหนด IP Address เป็น 192.168.0.3 ในส่วนนี้จะติดตั้งโปรแกรม process_send_file.exe ดังภาพ 23



ภาพ 23 โปรแกรม process_send_file.exe

โดยโปรแกรมจะตรวจสอบและทำงานวนซ้ำ ๆ ตามเวลาที่ตั้งค่าไว้ เวลาตรงนี้สามารถตั้งค่าเพิ่มหรือลดได้ขึ้นอยู่กับความเหมาะสมว่าจะให้โปรแกรมทำงานบ่อยแค่ไหนใน

การตรวจสอบคิวเตรียมไฟล์สำหรับส่งต่อไปยัง Raspberry Pi4 สาเหตุที่ไม่ตั้งเวลาให้โปรแกรมทำงานถี่หรือบ่อยเกินไปเพราะอาจจะทำให้โปรแกรมแอสคริปต์ระบบจะหยุดทำงานได้ โปรแกรมมีขั้นตอนการทำงานโดยเริ่มจากการอ่าน Copy queue. จากฐานข้อมูลว่ามีรายการที่ถูกสร้างขึ้นโดยโปรแกรม Nano data transfer หรือไม่ ถ้ามีให้ตรวจสอบว่ามี File หรือ Directory ผลการวิเคราะห์นั้นอยู่ใน path ตามที่อ่านได้จาก Copy queue. ถ้ามี File หรือ Directory จริงใช้โปรแกรม 7zip บีบอัดและแบ่ง File หรือ Directory นั้นออกเป็นไฟล์ย่อยเก็บไว้ใน Directory Temp_file/7Z_FILE ดังภาพ 24

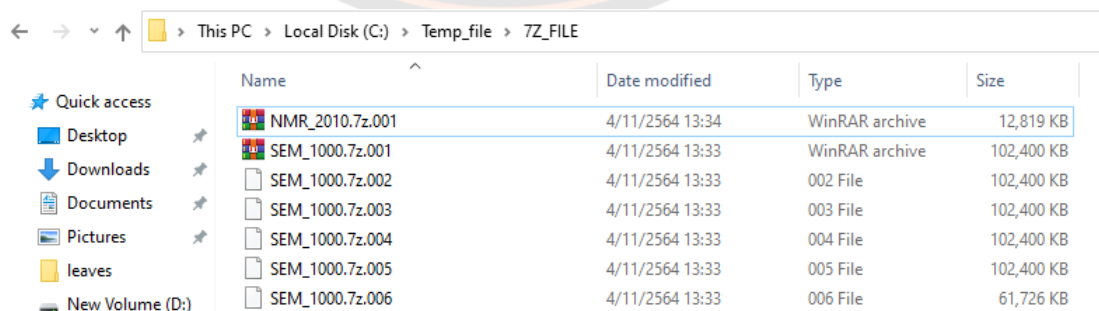
```

if (len(rows) != 0):
    for id, fileIntext in rows:
        fname = fileIntext.strip()
        p = fname.split("|")
        pathFile = path_tmp_file + "\\7z_FILE\\" + p[1] + ".7z"
        if path.exists(pathFile) == False:
            tic()
            print("+++++++Start 7zip encrypt+++++++")
            new_name = p[1].replace(" ", "")
            os.system(path_7z + "7za -v550m a -t7z \"" + path_tmp_file + "\\7z_FILE\\" + new_name + ".7z\" \" + Path_watchdog + \"")
            output = sp.getoutput(path_7z + "7za l \" + path_tmp_file + "\\7z_FILE\\" + new_name + ".7z.001")
            start = "Physical Size ="
            end = "Total"
            s = (output.split(start))[1].split(end)[0]
            y = s.split("\n")
            delete_empty = [ele for ele in y if ele.strip()]
            z7_volume = delete_empty[1].split(" = ")
            print("+++++++End 7zip encrypt+++++++")

```

ภาพ 24 ตัวอย่างการบีบอัดและแบ่งไฟล์ด้วยโปรแกรม 7zib ที่ 550MB

ในการแบ่งไฟล์ออกเป็นไฟล์ย่อยจะต้องหาขนาดการแบ่งที่เหมาะสมเพื่อโอนถ่ายไฟล์ให้ได้เต็มประสิทธิภาพกับเวลาที่เปิด Network interface card และ ไฟล์ที่ได้จะมีนามสกุล .7z.001.ไปจนถึง .7z.xxx ตัวอย่างดังภาพ 25



Name	Date modified	Type	Size
NMR_2010.7z.001	4/11/2564 13:34	WinRAR archive	12,819 KB
SEM_1000.7z.001	4/11/2564 13:33	WinRAR archive	102,400 KB
SEM_1000.7z.002	4/11/2564 13:33	002 File	102,400 KB
SEM_1000.7z.003	4/11/2564 13:33	003 File	102,400 KB
SEM_1000.7z.004	4/11/2564 13:33	004 File	102,400 KB
SEM_1000.7z.005	4/11/2564 13:33	005 File	102,400 KB
SEM_1000.7z.006	4/11/2564 13:33	006 File	61,726 KB

ภาพ 25 ภาพผลตัวอย่างการบีบอัดและแบ่งไฟล์ด้วยโปรแกรม 7zip ที่ 100MB

จำนวนไฟล์ขึ้นอยู่กับข้อกำหนดขนาดการแบ่ง การที่กำหนดขนาดแบ่งไฟล์เล็กเกินไปจะทำให้ได้ไฟล์จำนวนมากโปรแกรมจะทำงานช้าลงเพราะต้องวนลูปมากขึ้นและการเขียนไฟล์ลงแหล่งเก็บข้อมูลปลายทางจะช้าลงเนื่องจากการสร้างไฟล์ในระบบไฟล์จำนวนไฟล์ที่น้อยจะสร้างเร็วกว่าจำนวนไฟล์ที่มากทั้ง ๆ ที่จำนวนไฟล์ที่มากมีขนาดไฟล์เล็กกว่าขนาดของไฟล์ที่มีจำนวนน้อยเช่นไฟล์ 10 ไฟล์ขนาดไฟล์ละ 10Mb จะมีความเร็วในการสร้างไฟล์ในแหล่งเก็บข้อมูลมากกว่าไฟล์ที่มีจำนวน 100 ไฟล์ไฟล์ละ 1Mb และได้ Output จาก Web server ช้าลงเนื่องจากในกรณีที่ถ่ายโอนไฟล์จาก Raspberry Pi4 ไปยัง Web server ไม่ครบทำให้ไม่สามารถถอดรหัสลับและรวมไฟล์ด้วยโปรแกรม 7zip ได้ พอเสร็จจากการบีบอัดด้วยโปรแกรม 7zip ก็จะนำไฟล์ที่ได้เริ่มทำการเข้ารหัสลับแบบ AES โดยใช้ library pyAesCrypt ดังตัวอย่างภาพ 26 เก็บไฟล์ที่เข้ารหัสแบบ AES ไว้ใน Directory Temp_file/AES_FILE โดยไฟล์ที่ได้จะมีนามสกุล .aes ดังตัวอย่างภาพ 27 เมื่อเสร็จจากการเข้ารหัสลับแบบ AES จะมีการเขียน logfile, Queue ในฐานข้อมูลและลบรายการนั้นออกจาก Copy queue

```
# encrypt
for i in range(1, int(z7_volume[1]) + 1):
    print(p[1] + ".7z."f"{i:03}")
    print("+++++++Start AES encrypt+++++++")
    with open(path_tmp_file + "\\7Z_FILE\\" + new_name + ".7z."f"{i:03}", "rb") as fIn:
        with open(path_tmp_file + "\\AES_FILE\\" + new_name + ".7z."f"{i:03}.aes", "wb") as fOut:
            pyAesCrypt.encryptStream(fIn, fOut, password, bufferSize)
    processT = tac()
    print("Encrypt succeeded, time consuming " + str(processT))
    print("+++++++END AES encrypt+++++++")
```


ภาพ 26 ตัวอย่างการเข้ารหัสลับแบบ AES ด้วย pyAesCrypt

Name	Date modified	Type	Size
NMR_2010.7z.001.aes	4/11/2564 13:34	AES File	12,819 KB
SEM_1000.7z.001.aes	4/11/2564 13:34	AES File	102,401 KB
SEM_1000.7z.002.aes	4/11/2564 13:34	AES File	102,401 KB
SEM_1000.7z.003.aes	4/11/2564 13:34	AES File	102,401 KB
SEM_1000.7z.004.aes	4/11/2564 13:34	AES File	102,401 KB
SEM_1000.7z.005.aes	4/11/2564 13:34	AES File	102,401 KB
SEM_1000.7z.006.aes	4/11/2564 13:34	AES File	61,727 KB

ภาพ 27 ผลตัวอย่างการเข้ารหัสลับแบบ AES ด้วย pyAesCrypt

และในส่วนของ File server ยังได้ติดตั้งโปรแกรม send_fileto_raspberry.exe ดังภาพ 28 ซึ่งเป็นโปรแกรมถ่ายโอนไฟล์ที่ถูกบีบอัดด้วยโปรแกรม 7zip และเข้ารหัสลับแบบ AES จาก

File Server ไป Raspberry Pi4 โดยมีขั้นตอนการทำงานเริ่มจากตรวจสอบ Network interface card Eth0 ว่ามี Connection หรือไม่ถ้าไม่มีก็ให้แสดงผล “Eth0 of raspberry pi is offline.....” และให้ทำการตรวจสอบต่อไปจนกว่าจะมี Connection .ให้เริ่มอ่าน Queue ว่ามีจำนวนเรคคอร์ดหรือไม่ถ้าไม่มีให้แสดงผล “No data in Temp_file and logfile on File server.....” แต่ถ้ามีให้ทำการเชื่อมต่อ Connection กับ Network interface card Eth0 ของ Raspberry Pi4 และเริ่มทำการถ่ายโอนข้อมูลไป Raspberry Pi4 ถ้าไฟล์ไหนถ่ายโอนสำเร็จทางฝั่งของ File server ให้ลบรายการนั้นออกจาก Queue และบันทึก log_time_send และลบไฟล์รายการนั้นทิ้งไปและทางฝั่งของ Raspberry Pi4 ให้เขียน Queue และ logfile รายการที่ส่งสำเร็จจาก File server ทำงานแบบนี้ไปเรื่อย ๆ จนกว่า Connection จะถูกตัดตัวอย่างดังภาพ 29

 send_fileto_raspberry.exe

ภาพ 28 โปรแกรม send_fileto_raspberry.exe

```

check_size = os.path.getsize(path_tmp_file + "\\AES_FILE\\" + df[2])
if check_size == int(df[1]):
    print("Transfer file " + df[2])
    sftp.put(path_tmp_file + "\\AES_FILE\\" + df[2] + ', raspberry_tmp_file + df[2] + ', callback=None, confirm=True)
    create_raspberry_queue(fname)
    listdel.append(fname)
    listdelfile.append(df[2])
    time_end = tac()
    t_end = time.time()
    t_use = t_end - t_start
    time_us = '%.2f' % (t_use)
    date_now = datetime.datetime.now()
    file_send = str(date_now) + " " + df[2] + "@" + str(time_us)
    create_fileservers_log_time_send(file_send)
    delete_fileservers_queue(listdel)
    os.system('del ' + path_tmp_file + "\\AES_FILE\\" + df[2])
    time.sleep(0.5)
    print("Delete file " + df[2])

time.sleep(1)
s.close()

```

ภาพ 29 ตัวอย่างการเขียนไฟล์เมื่อถ่ายโอนไฟล์สำเร็จ

- ส่วนของ Raspberry Pi4 จะมี Network interface card 2 อันคือ Eth0 และ Eth1 โดย Network interface card Eth0 จะเชื่อมต่อเข้ากับ Network interface card Eth1 ของ File server กำหนด IP Address เป็น 192.168.0.4 และ Network interface card Eth1 จะเชื่อมต่อกับ Network ภายนอกกับ IP Address แบบอัตโนมัติจาก DHCP Server การทำงานในส่วนนี้จะติดตั้งโปรแกรม send_file_service.py ทำหน้าที่ควบคุมการเปิดปิด

Network interface และส่งไฟล์ผลการวิเคราะห์ที่ถูกบีบอัด แบ่งไฟล์ด้วยโปรแกรม 7zip และเข้ารหัสลับแบบ AES ไปยัง Web server โดยขั้นตอนการทำงานของโปรแกรมจะเริ่มทำการปิดการทำงานของ Network interface card Eth0 ที่เชื่อมต่อกับ File Server และทำการเปิดการทำงานของ Network interface card Eth1 ที่เชื่อมต่อกับ Web server ตามเวลาที่ตั้งค่าไว้หลังจากปิดการทำงานของ Network interface card Eth0 และเปิด Network interface card Eth1 ให้อ่านคิวการทำงานว่ามีคิวหรือไม่ ถ้าไม่มีให้แสดงผล “No data in queue raspberry pi....” แต่ถ้ามีให้ทำการ ssh เชื่อมต่อกับ Web server และถ่ายโอนไฟล์ด้วยโปรโตคอล sftp ตามคิวถ้าไฟล์รายการไหนถ่ายโอนสำเร็จให้ทางฝั่งของ Raspberry Pi4 ให้นำรายการนั้นออกจาก Queue.เขียน log_time_send และลบไฟล์รายการนั้นทิ้งไป และทางฝั่งของ Web server ให้เขียน Queue รายการที่ส่งสำเร็จจาก Raspberry Pi4 ทำงานแบบนี้ไปเรื่อย ๆ จนกว่า Connection จะถูกตัดตามเวลาที่ได้ตั้งค่าไว้

4. ส่วนของ Web Server จะมี Network interface card 1 อันคือ Eth0 กำหนด IP Address เป็น 10.31.46.12

5. ขนาดไฟล์ที่ใช้ในการทดลอง

จากการสำรวจข้อมูลผลการวิเคราะห์ด้วยเครื่องมือวิทยาศาสตร์ในช่วงเวลา 6 ปีย้อนหลัง (พ.ศ. 2561 – พ.ศ. 2566) จากจำนวนเครื่องมือวิทยาศาสตร์ทั้งหมด 22 รายการ สามารถสรุปรายละเอียดการส่งข้อมูลผลการวิเคราะห์ของเครื่องมือวิทยาศาสตร์ประกอบด้วยจำนวนคิวของการให้บริการ จำนวนตัวอย่างที่ส่งตรวจวิเคราะห์ และขนาดไฟล์ทั้งหมดที่ทำการส่งให้กับผู้รับบริการของแต่ละเครื่องมือได้ดังที่แสดงในตารางที่ 2

ตาราง 2 ข้อมูลผลการวิเคราะห์ด้วยเครื่องมือวิทยาศาสตร์ 6 ปีย้อนหลัง

เครื่องมือวิทยาศาสตร์	พ.ศ. 2561 – พ.ศ. 2566		ขนาดไฟล์ที่ส่งทั้งหมด
	จำนวนคิว	จำนวนตัวอย่าง	
AAS	22	1795	1GB
CAA	34	34	1GB
SFM-4	63	564	1GB
DFLS	12	53	1GB
GFAAS	10	1920	1GB
ICP-OES	17	2310	1GB

Titan-MPS	2	15	1GB
GC-MS/MSTQ	16	16	1GB
FESEM	115	905	1GB
HPLC-1100	74	74	600MB
HPLC-1260	73	73	600MB
SEM	208	1454	500MB
Instron-5965	112	112	200MB
NMR	621	621	100MB
FT-IR	245	245	100MB
DSC	33	133	100MB
LCMS	328	328	100MB
ZS	35	253	10MB
MR	154	3958	10MB
UV-VS	30	665	10MB
XRD-BRUKER	251	1010	10MB
BET	136	342	1MB

จากตาราง 2 สามารถจำแนกเครื่องมือตามช่วงของขนาดไฟล์ผลการวิเคราะห์ที่ทำการส่งให้
ผู้รับบริการได้ 3 ช่วงซึ่งแสดงในตารางที่ 3

ตาราง 3 ข้อมูลข้อมูลจำนวนเครื่องมือวิทยาศาสตร์จำแนกช่วงขนาดของไฟล์

ลำดับ	ช่วงขนาดไฟล์	จำนวน เครื่องมือ	จำนวนตัวอย่าง ทั้งหมด	จำนวนตัวอย่าง เฉลี่ยต่อ 1 วัน
1	ต่ำกว่า 500 MB	10	7,666	33
2	ระหว่าง 500 MB ถึง 1 GB	12	9,213	40
3	สูงกว่า 1 GB	2	362	2

จากตาราง 3 แสดงข้อมูลจำนวนเครื่องมือวิทยาศาสตร์จำแนกช่วงขนาดของไฟล์ที่ส่งให้ผู้รับบริการ จะเห็นได้ว่าจำนวนเครื่องมือที่อยู่ในช่วงขนาดไฟล์ระหว่าง 500 MB ถึง 1 GB มีจำนวนมากที่สุดที่ 12 เครื่อง และมีจำนวนการส่งตัวอย่างเฉลี่ยต่อ 1 วันสูงที่สุดอยู่ที่ 40 ตัวอย่าง ผู้วิจัยจึงเลือกใช้ตัวอย่างไฟล์ขนาดที่ 1 GB เป็นขนาดไฟล์ที่สูงที่สุดสำหรับการทดลอง ซึ่งเป็นขนาดไฟล์ที่มีค่าเฉลี่ยในการใช้งานสำหรับการส่งตัวอย่างการทดลองสำหรับเครื่องมือวิทยาศาสตร์มากที่สุดนั่นเอง

6. ขั้นตอนการทดลอง

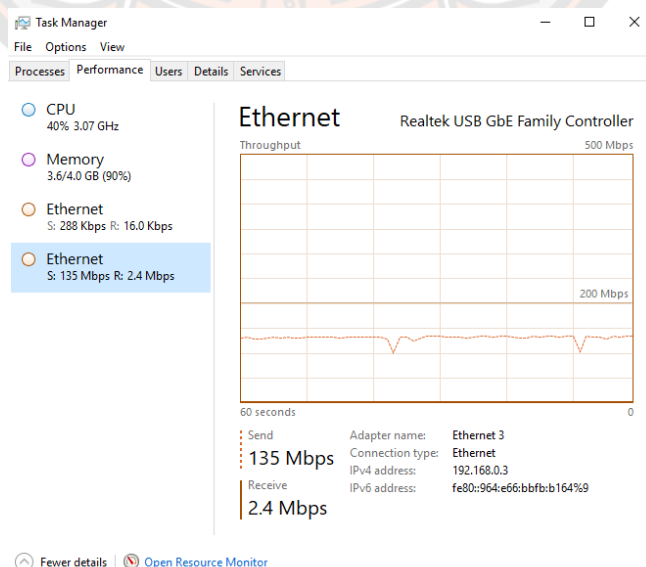
1. ใช้โปรแกรม 7zip แบ่งไฟล์ 1000MB เป็นไฟล์ขนาดเล็กลงโดยเริ่มแบ่งจาก 10MB ไปจนถึง 1000MB พร้อมเข้ารหัสแบบ AES
2. ตั้งค่าให้ Raspberry Pi4 เปิด interface network ทุก 55, 60, 65, 70, 75 วินาที
3. บันทึกเวลาโอนถ่ายข้อมูลแต่ละไฟล์ที่โอนถ่ายสำเร็จ หาเวลารวม และเวลาเฉลี่ย
4. หาขนาดการแบ่งไฟล์และเวลาเปิด ปิด interface network ที่ทำเวลาเฉลี่ยโอนถ่ายข้อมูลดีที่สุด

ในการทดลองการโอนถ่ายไฟล์ข้อมูลจาก File Server ไปยัง Raspberry Pi4 และจาก Raspberry Pi4 ไปยัง Web server จะเป็นการโอนถ่ายข้อมูลแบบ Half-duplex ดังภาพ 30 เป็นการโอนถ่ายไฟล์ข้อมูลจาก File server ไป Raspberry Pi4 ผ่านทาง USB Gigabit ethernet adapter ไปยัง Local ethernet adapter ของ Raspberry Pi4

7. ข้อมูลด้านฮาร์ดแวร์และระบบปฏิบัติการ

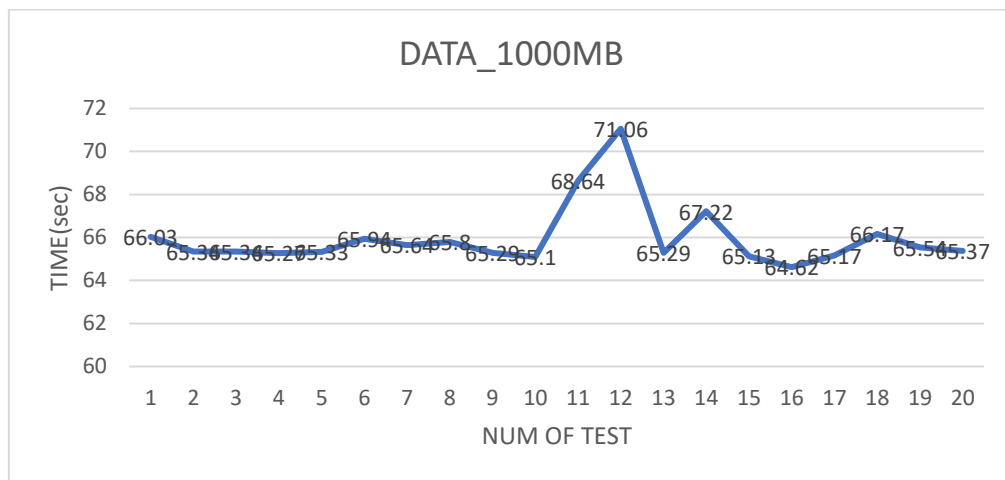
1. File server
 - Intel(R) Core(TM) i3-2100 CPU @3.10GHz 3.10 GHz
 - RAM 4 GB
 - HARD DISK 300GB 7200 rpm
 - OS Windows server 2019
2. Raspberry Pi4 Model B
 - Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz

- 4GB LPDDR4-2400 SDRAM (depending on model)
 - 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE Gigabit Ethernet
 - 2 USB 3.0 ports; 2 USB 2.0 ports.
 - Raspberry Pi standard 40 pin GPIO header (fully backwards compatible with previous boards)
 - 2 x micro-HDMI ports (up to 4kp60 supported)
 - 2-lane MIPI DSI display port
 - 2-lane MIPI CSI camera port
 - 4-pole stereo audio and composite video port
 - H.265 (4kp60 decode), H264 (1080p60 decode, 1080p30 encode) OpenGL ES 3.0 graphics
 - Micro-SD card slot for loading operating system and data storage
3. Web server
- Intel(R) Core(TM) i3-2100 CPU @3.10GHz 3.10 GHz
 - RAM 4 GB
 - HARD DISK 300GB 7200 rpm
 - OS Ubuntu 20.04



ภาพ 30 ความเร็วโอนถ่ายไฟล์ข้อมูลจาก File server ไป Raspberry Pi4

ความเร็วในการโอนถ่ายข้อมูลจาก File server ไป Raspberry Pi4 ผ่านทาง USB Gigabit Ethernet adapter ไปยัง Local Ethernet adapter ของ Raspberry Pi4 สูงสุดอยู่ที่ 135 Mbps แต่การโอนถ่ายจริงๆ ความเร็วจะไม่คงที่ มีขึ้นมีลงตามการประมวลผลของ CPU ของ Raspberry Pi4 และประสิทธิภาพของเครื่อง File server จึงทำให้เวลารวมของการโอนถ่ายข้อมูลไม่คงที่ดังตัวอย่างภาพ 31 เป็นกราฟแสดงเวลาโอนถ่ายข้อมูล 1000MB 20 ครั้งจาก File Server ผ่านทาง USB gigabit ethernet adapter ไปยัง Local ethernet adapter ของ Raspberry Pi4

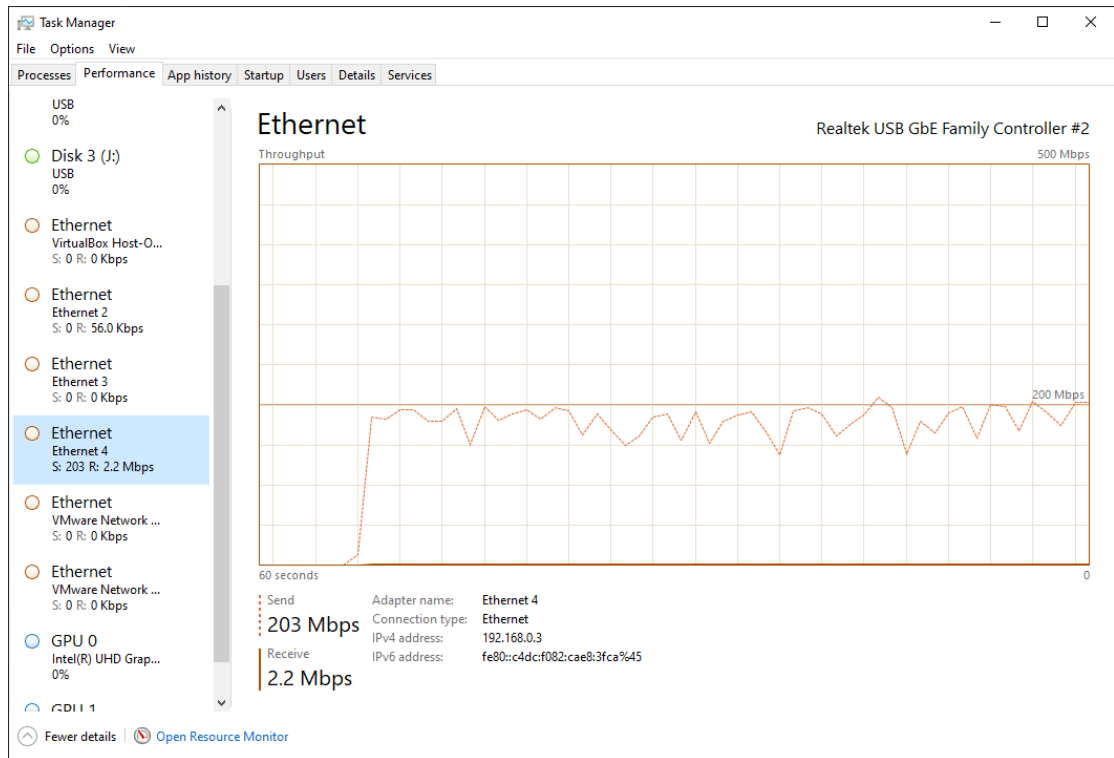


ภาพ 31 เป็นกราฟแสดงเวลาโอนถ่ายข้อมูล 1000MB 20 ครั้ง จาก File server ไป Raspberry Pi4

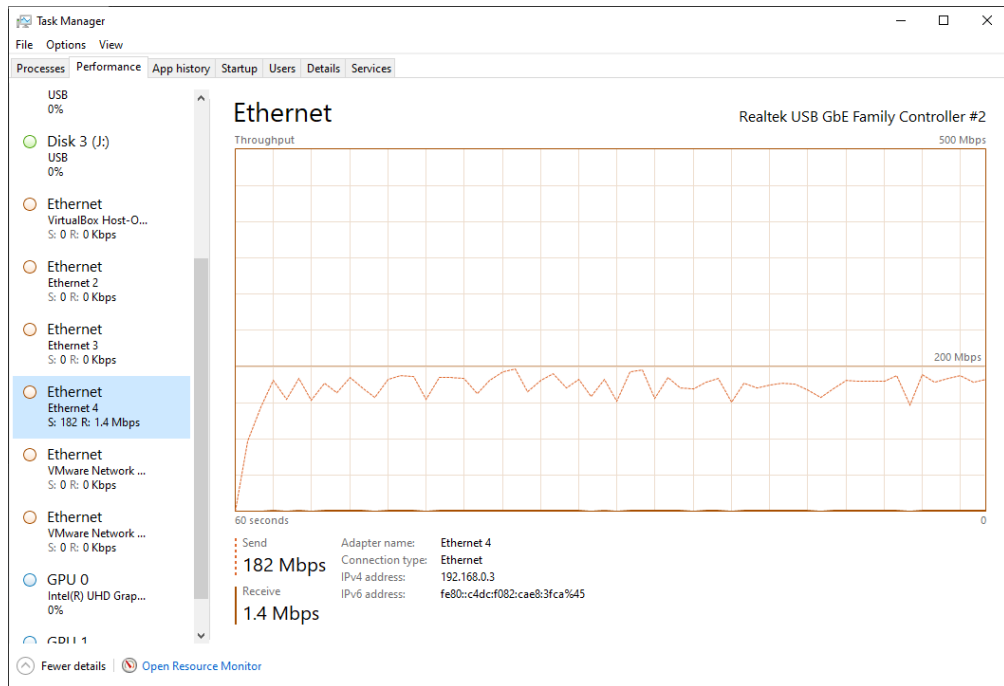
ทดลองเปลี่ยนเครื่อง File server เป็นเครื่องที่มีประสิทธิภาพดีขึ้นดังนี้

- Intel(R) Core (TM) i7-8700 CPU @3.20GHz 3.19 GHz
- RAM 16 GB
- HARD DISK SSD 450GB
- OS Windows 10 64 bit

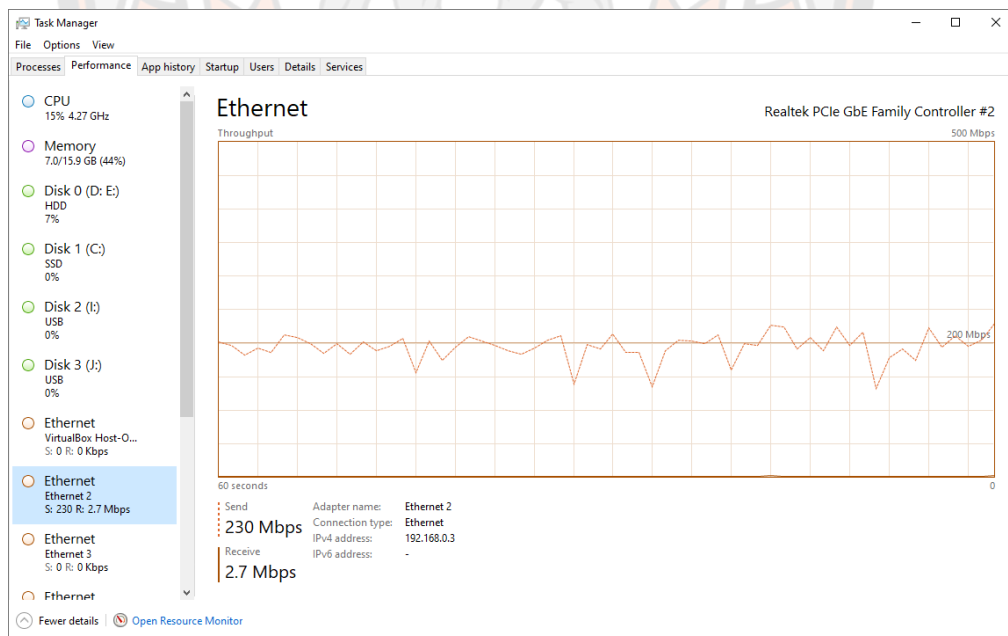
และทำการทดสอบโอนถ่ายข้อมูลจาก File server ไป Raspberry Pi4



ภาพ 32 กราฟแสดงเวลาโอนถ่ายข้อมูลจาก USB Gigabit Ethernet adapter ของ File server ไปยัง Local Ethernet adapter ของ Raspberry Pi4

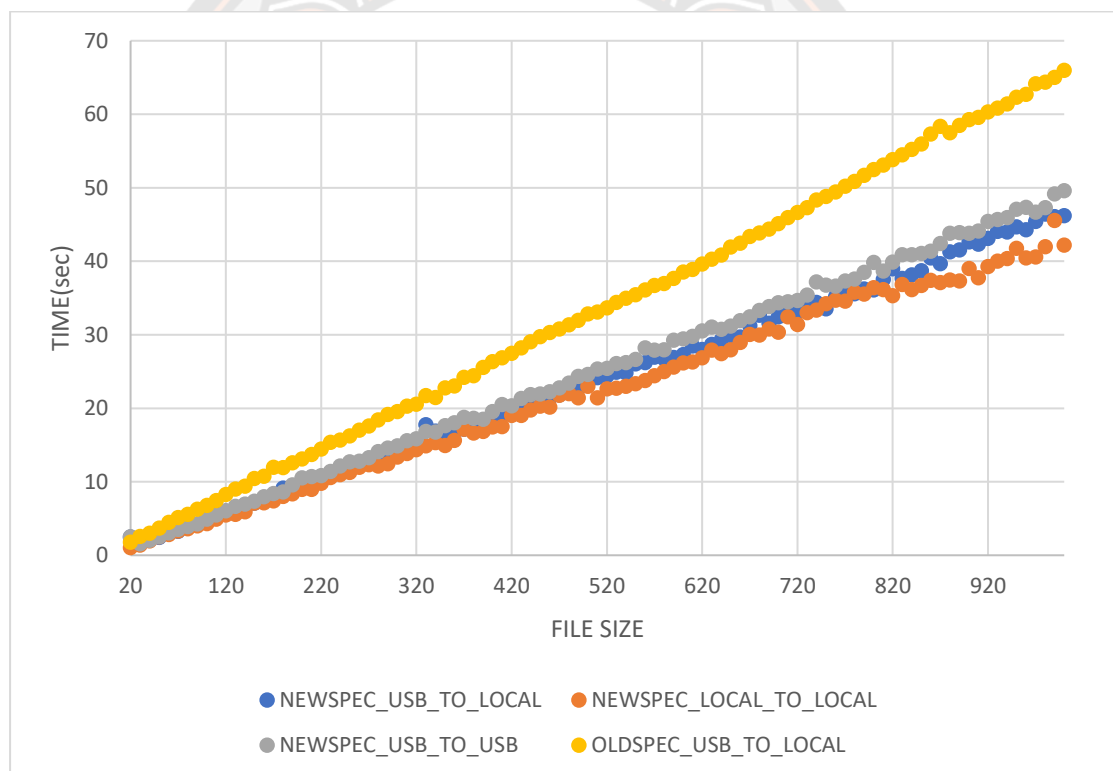


ภาพ 33 กราฟแสดงเวลาโอนถ่ายข้อมูลจาก USB Gigabit Ethernet adapter ของ File server ไปยัง USB Gigabit Ethernet adapter ของ Raspberry Pi4



ภาพ 34 กราฟแสดงเวลาโอนถ่ายข้อมูลจาก Local Ethernet adapter ของ File server ไปยัง Local Ethernet adapter ของ Raspberry Pi4

จากภาพ 32 33 และ 34 ตามลำดับเป็นการโอนถ่ายข้อมูลจาก USB Gigabit Ethernet adapter ของ File server ไปยัง Local Ethernet adapter ของ Raspberry Pi4 ทำความเร็วได้ประมาณ 203 Mbps การโอนถ่ายข้อมูลจาก USB Gigabit Ethernet adapter ของ File server ไปยัง USB Gigabit Ethernet adapter ของ Raspberry Pi4 ทำความเร็วได้ประมาณ 182 Mbps และการโอนถ่ายข้อมูลจาก Local Ethernet adapter ของ File server ไปยัง Local Ethernet adapter ของ Raspberry Pi4 ทำความเร็วได้ประมาณ 230 Mbps ถ้าเปรียบเทียบความเร็วการโอนถ่ายข้อมูลจาก USB Gigabit Ethernet adapter ของ File server ไปยัง Local Ethernet adapter ของ Raspberry Pi4 พบว่าสเปค Hardware ของ File server ที่แตกต่างกันจะมีผลต่อความเร็วโอนถ่ายข้อมูลดังภาพ 35



ภาพ 35 กราฟแสดงเวลาโอนถ่ายข้อมูลจาก File server ไปยัง Raspberry Pi4

จากการทดสอบนี้สรุปได้ว่าประสิทธิภาพของเครื่องมีผลต่อการโอนถ่ายข้อมูล แต่ผลการทดลองที่ได้นำเสนอจะอ้างอิงความเร็วดังภาพ 30 ซึ่งเป็นความเร็วโอนถ่ายไฟล์ข้อมูลจาก File server สเปคต่ำ ไป Raspberry Pi4

8. การประเมินและวัดผลการทดลอง

1. ทดลองหาความสัมพันธ์ขนาดไฟล์กับเวลาในการโอนถ่ายข้อมูลให้สอดคล้องกับการตั้งเวลาเปิดปิด Interface network ของ Raspberry Pi4 เชื่อมต่อกับ File server โดยการปรับขนาดของไฟล์ ปรับเวลาเปิดปิด Interface network และบันทึกเวลา
 - 1.1 สมมติฐาน
 1. เวลาในการเปิด Interface network ของ Raspberry Pi4 เชื่อมต่อกับ File server มีค่าเฉลี่ยเวลาในการโอนถ่ายข้อมูลแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ที่ระดับนัยสำคัญ 0.05
 2. ปริมาณของจำนวนไฟล์ที่แตกต่างกันใช้ระยะเวลาโอนถ่ายข้อมูลแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ที่ระดับนัยสำคัญ 0.05
 3. เวลาในการเปิด Interface network ของ Raspberry Pi4 ขนาดไฟล์และจำนวนข้อมูลมีผลต่อระยะเวลาโอนถ่ายข้อมูลอย่างมีนัยสำคัญทางสถิติ ที่ระดับนัยสำคัญ 0.05
2. ทดลองจำลองสถานการณ์การโดยการ Scan port และยิง Packet เข้าระบบและดูว่าโปรแกรม Snort สามารถตรวจจับ Packets ได้กี่เปอร์เซ็นต์และระบบสามารถจัดเก็บ log packets ที่ Firewall ตรวจสอบและ Block ได้กี่เปอร์เซ็นต์
3. วัดร้อยละความถูกต้องของข้อมูลผลการวิเคราะห์ที่ได้รับผ่านทาง website

บทที่ 4

ผลการวิจัย

1. ผลการทดลองหาความสัมพันธ์ขนาดไฟล์กับเวลาในการโอนถ่ายข้อมูลให้สอดคล้องกับการตั้งเวลาเปิดปิด Interface network ของ Raspberry Pi4 เชื่อมต่อกับ File server

การวิจัยเพื่อหาความสัมพันธ์ขนาดไฟล์กับเวลาในการโอนถ่ายข้อมูลให้สอดคล้องกับการตั้งเวลาเปิดปิด Interface network ของ Raspberry Pi4 เชื่อมต่อกับ File server โดยการปรับขนาดไฟล์ปรับเวลาเปิดปิด Interface network เพื่อให้ได้ขนาดไฟล์ที่เหมาะสมกับเวลาเปิดปิด Interface ที่ทำเวลาในการโอนถ่ายข้อมูลได้ดีที่สุด โดยการทดลองแบ่งไฟล์ขนาด 1000Mb เป็นไฟล์ขนาดต่างๆ ตั้งแต่ 10Mb ถึง 1000Mb แล้วถ่ายโอนไปยัง Raspberry Pi4 ที่เวลาเปิดปิด Interface network ของ Raspberry Pi ที่เวลา 55, 60, 65, 70 และ 75 วินาที แล้วหาเวลาโอนถ่ายข้อมูลเฉลี่ย ข้อมูลที่ได้จากการทดลองนำมาทำการวิเคราะห์การถดถอยเชิงพหุคูณ (Multiple regression analysis), เพื่อทดสอบความสัมพันธ์ของตัวแปรอิสระหลายตัวที่มีผลต่อตัวแปรตามเพียงตัวเดียว ใช้ระดับความเชื่อมั่น 95% ความน่าจะเป็นสำหรับบอกค่านัยสำคัญทางสถิติ (α) ในงานวิจัยครั้งนี้กำหนดค่าไว้ที่ 0.05 ตามสมมุติฐานของการวิจัย

สัญลักษณ์ที่ใช้ในการวิเคราะห์ข้อมูล

p	แทน	ระดับของความมีนัยสำคัญ
t	แทน	ค่าทดสอบของนัยสำคัญของค่าเฉลี่ย 2 กลุ่ม (ค่าสถิติ t-test)
R	แทน	ค่าสัมประสิทธิ์สัมพันธ์ระหว่างตัวแปรอิสระ (Multiple R)
R square	แทน	ค่าสัมประสิทธิ์การพยากรณ์
Adjusted R square	แทน	ค่าสัมประสิทธิ์การพยากรณ์เมื่อปรับแล้ว
B	แทน	ค่าสัมประสิทธิ์การถดถอยของตัวพยากรณ์ในรูปคะแนนดิบ
Beta	แทน	ค่าสัมประสิทธิ์ถดถอยของตัวพยากรณ์ในรูปคะแนนมาตรฐาน เป็นค่าที่แสดงน้ำหนักของความสัมพันธ์หรืออิทธิพลของตัวแปรอิสระแต่ละตัวที่มี ต่อตัวแปรตาม
SE (est.)	แทน	ค่าความคลาดเคลื่อนมาตรฐานจากการประมาณค่าที่เกิดจากการ ถดถอยพหุคูณ
Sig.	แทน	ความน่าจะเป็นสำหรับบอกค่านัยสำคัญทางสถิติ (Significance)
*	แทน	นัยสำคัญทางสถิติในการวิจัยครั้งนี้ กำหนดระดับนัยสำคัญ (α) ไว้ที่ 0.05

ตาราง 4 ผลการทดสอบสมมติฐาน โดยใช้ Multiple regression analysis ในการวิเคราะห์
หาเวลาในการโอนถ่ายข้อมูล

ปัจจัยด้านขนาด	Unstandardized		Standardized	t	Sig.
ไฟล์ จำนวน	coefficients		coefficients		
ไฟล์และเวลา	B	SE	Beta		
เปิดปิด					
Interface					
network					
ค่าคงที่	61.110	0.488		125.288	0.000*
เวลาเปิด ปิด	0.0555	0.007	0.174	7.486	0.000*
Interface					
network ของ					
Raspberry Pi4					
จำนวนไฟล์ที่ได้	0.1615	0.005	0.891	33.885	0.000*
จากการแบ่ง					
ขนาดไฟล์ที่แบ่ง	0.0005	0.000	0.061	2.314	0.021*
R	0.8766				
R square	0.7684				
Adjusted R	0.77				
square					
SE (est.)	1.08				
Durbin-Watson	2.02				

* Pvalue \leq 0.05

จากการวิเคราะห์ตารางที่ 4 การทดสอบเงื่อนไขของการวิเคราะห์การถดถอยเชิงพหุคูณ (Multiple regression analysis) พบว่า ปัจจัยที่มีอิทธิพลต่อตัวแปรตาม “เวลาในการโอนถ่ายข้อมูล” คือ จำนวนไฟล์ที่ได้จากการแบ่ง เวลาในการเปิดปิด Interface network ของ Raspberry Pi4 และขนาดไฟล์ที่แบ่ง โดยจำนวนไฟล์จะมีความสัมพันธ์กับขนาดไฟล์ที่แบ่ง ที่ Sig. เท่ากับ 0.000, 0.000 และ 0.021 ตามลำดับซึ่งน้อยกว่านัยสำคัญทางสถิติที่ระดับ 0.05 ค่าสัมประสิทธิ์สหสัมพันธ์ R ระหว่างตัวแปรอิสระ “ขนาดไฟล์ จำนวนไฟล์และเวลาเปิดปิด Interface network” และตัวแปรตาม “เวลา

ในการโอนถ่ายข้อมูล” ซึ่งเท่ากับ 0.8766 ดังนั้น ขนาดไฟล์ จำนวนไฟล์และเวลาเปิดปิด Interface network กับ เวลาในการโอนถ่ายข้อมูล มีความสัมพันธ์กันโดยมีค่า R square เท่ากับ 0.7684 หรือเท่ากับร้อยละ 76.84

สมมติฐานข้อที่ 1 เวลาในการเปิด Interface network ของ Raspberry Pi4 เชื่อมต่อกับ File server มีค่าเฉลี่ยเวลาในการโอนถ่ายข้อมูลแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ที่ระดับนัยสำคัญ 0.05

ผลการทดสอบสมมติฐานด้วยการวิเคราะห์การถดถอยพหุคูณ (Multiple regression analysis) ตัวแปรอิสระ ด้านเวลาเปิด ปิด Interface network ของ Raspberry Pi4 (X_2) มีค่า Sig. เท่ากับ 0.000 ซึ่งน้อยกว่านัยสำคัญทางสถิติที่ระดับ 0.05 มีค่า B เท่ากับ 0.0555 และ Beta เท่ากับ 0.174 หมายความว่าปัจจัยด้านเวลาเปิด ปิด Interface network ของ Raspberry Pi4 (X_2) มีความสำคัญหรืออิทธิพลต่อตัวแปรตาม “เวลาในการโอนถ่ายข้อมูล” อย่างมีนัยสำคัญทางสถิติ

สมมติฐานข้อที่ 2 ปริมาณของจำนวนไฟล์ที่แตกต่างกันใช้ระยะเวลาโอนถ่ายข้อมูลแตกต่างกันอย่างมีนัยสำคัญทางสถิติ ที่ระดับนัยสำคัญ 0.05

ผลการทดสอบสมมติฐานด้วยการวิเคราะห์การถดถอยพหุคูณ (Multiple regression analysis) ตัวแปรอิสระ ด้านจำนวนไฟล์ที่ได้จากการแบ่ง (X_1) มีค่า Sig. เท่ากับ 0.000 ซึ่งน้อยกว่านัยสำคัญทางสถิติที่ระดับ 0.05 มีค่า B เท่ากับ 0.1615 และ Beta เท่ากับ 0.891 หมายความว่าปัจจัยด้านจำนวนไฟล์ที่ได้จากการแบ่ง (X_1) มีความสำคัญหรืออิทธิพลต่อตัวแปรตาม “เวลาในการโอนถ่ายข้อมูล” อย่างมีนัยสำคัญทางสถิติ

สมมติฐานข้อที่ 3 เวลาในการเปิด Interface network ของ Raspberry Pi4 ขนาดไฟล์และจำนวนข้อมูลมีผลต่อระยะเวลาโอนถ่ายข้อมูลอย่างมีนัยสำคัญทางสถิติ ที่ระดับนัยสำคัญ 0.05

ผลการทดสอบสมมติฐานด้วยการวิเคราะห์การถดถอยพหุคูณ (Multiple regression analysis) ให้ค่าสัมประสิทธิ์สหสัมพันธ์ (R) ระหว่างตัวแปรอิสระ คือขนาดไฟล์ จำนวนไฟล์และเวลาเปิดปิด Interface network ด้านจำนวนไฟล์ที่ได้จากการแบ่ง (X_1) มีค่า Sig. เท่ากับ 0.000 ซึ่งน้อยกว่านัยสำคัญทางสถิติที่ระดับ 0.05 มีค่า B เท่ากับ 0.1615 และ Beta เท่ากับ 0.891 ด้านเวลาเปิดปิด Interface network ของ Raspberry Pi4 (X_2) มีค่า Sig. เท่ากับ 0.000 ซึ่งน้อยกว่านัยสำคัญทางสถิติที่ระดับ 0.05 มีค่า B เท่ากับ 0.0555 และ Beta เท่ากับ 0.174 ด้านขนาดไฟล์ที่แบ่ง (X_3) มีค่า Sig. เท่ากับ 0.021 ซึ่งน้อยกว่านัยสำคัญทางสถิติที่ระดับ 0.05 มีค่า B เท่ากับ 0.0005 และ Beta เท่ากับ 0.061 ตัวแปรตามคือ เวลาในการโอนถ่ายข้อมูล มีค่า R เท่ากับ 0.8766 หมายความว่า ปัจจัยด้านขนาดไฟล์ จำนวนไฟล์และเวลาเปิดปิด Interface network มีผลต่อเวลาในการโอนถ่ายข้อมูล ร้อยละ 87.66

ดังนั้นตัวแปรอิสระที่สามารถนำไปใช้พยากรณ์ได้ คือ ด้านจำนวนไฟล์ที่ได้จากการแบ่ง (X_1) ด้านเวลาเปิด ปิด Interface network ของ Raspberry Pi4 (X_2) และ ด้านขนาดไฟล์ที่แบ่ง (X_3) สามารถเขียนสมการพยากรณ์เวลาที่ใช้ในการโอนถ่ายข้อมูลเมื่อนำตัวแปรจำนวนไฟล์ที่ได้จากการแบ่ง (X_1) เวลาเปิด ปิด Interface network ของ Raspberry Pi4 (X_2) และ ขนาดไฟล์ที่แบ่ง (X_3) เข้าสมการของ Multiple linear regression ดังนี้

จากผลการทดลองสามารถเขียนสมการทดลองได้ดังนี้

$$\hat{Y} = 61.1106 + 0.1615(X_1) + 0.0555(X_2) + 0.0005(X_3)$$

โดย \hat{Y} คือเวลาที่ใช้ในการโอนถ่ายข้อมูล

(X_1) คือจำนวนไฟล์ข้อมูล

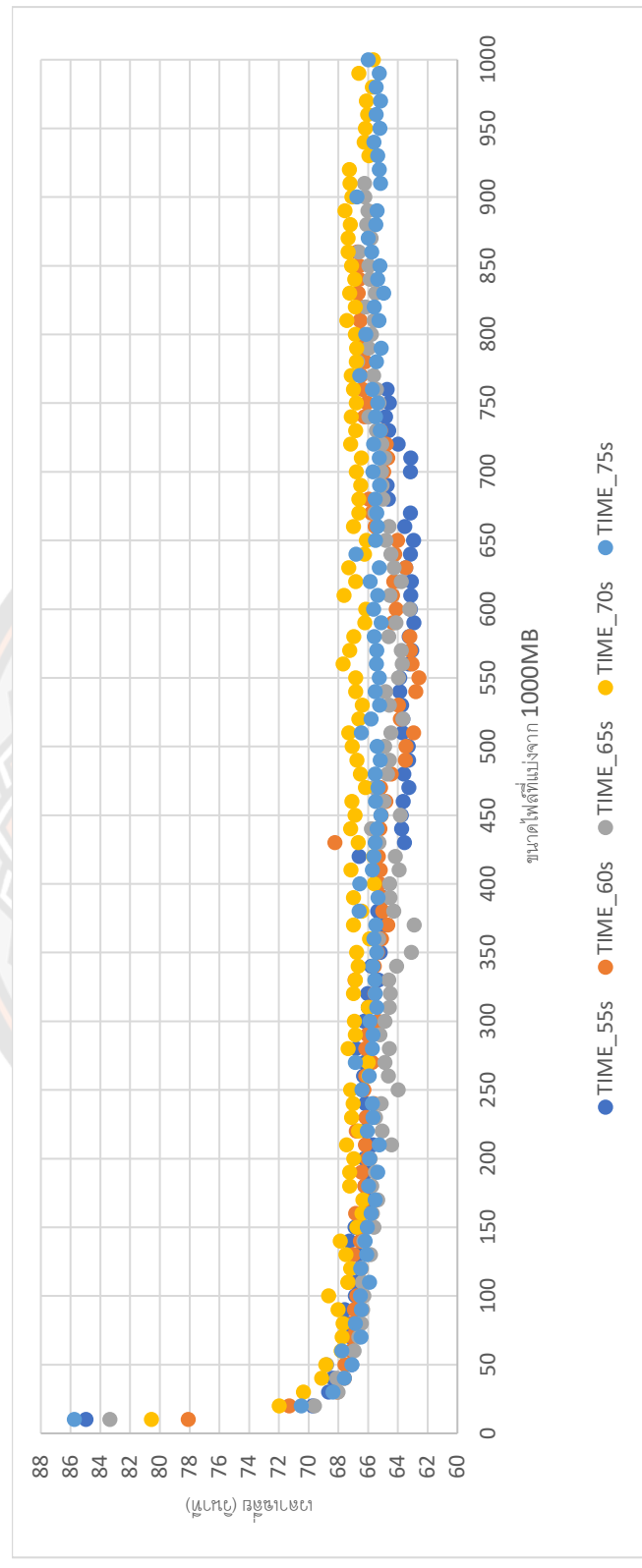
(X_2) คือเวลาในการเปิดปิด Interface network ของ Raspberry Pi4

(X_3) คือขนาดของไฟล์

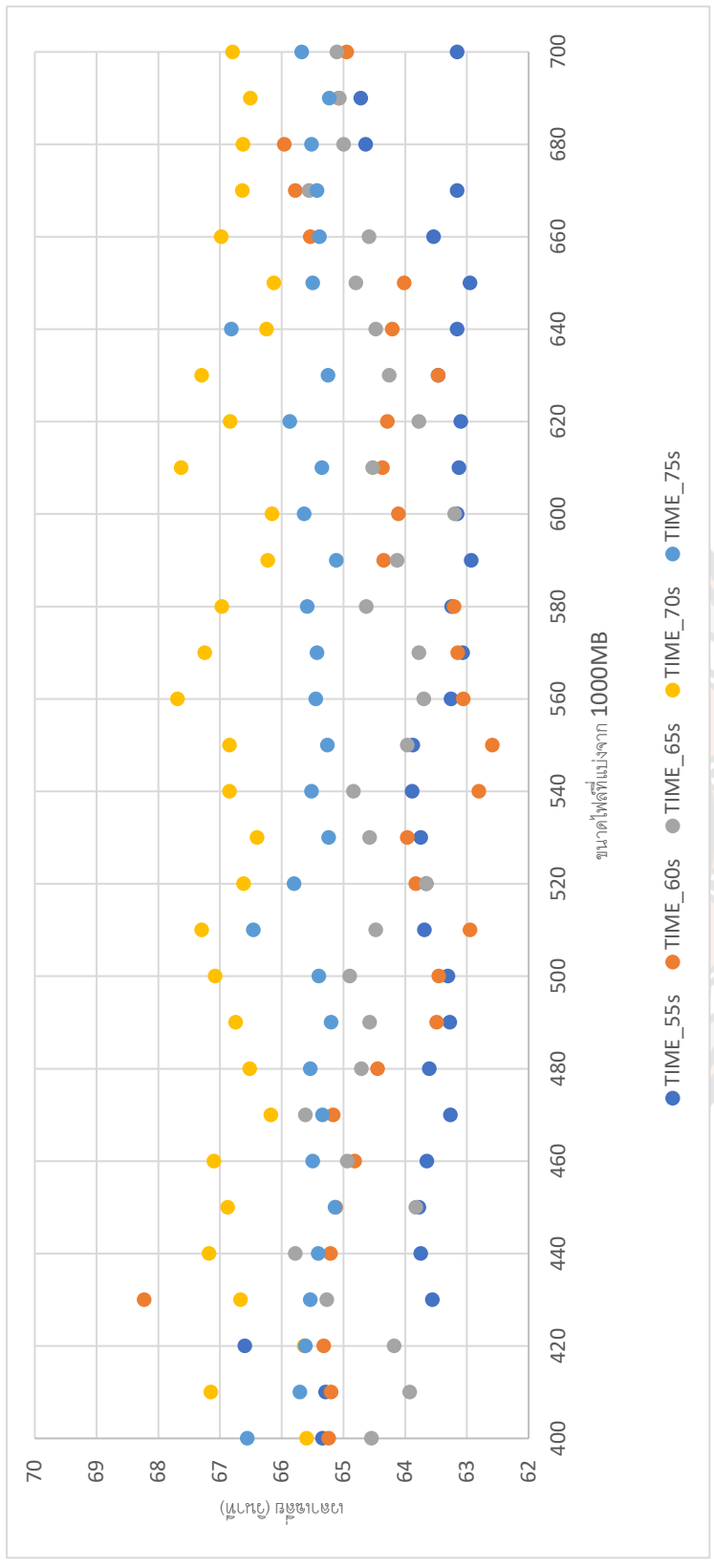
สมการนี้สามารถใช้พยากรณ์หาเวลาที่ใช้ในการโอนถ่ายข้อมูลมีความแม่นยำที่ 76.84%

การทดลองนี้เป็นการทดลองแสดงให้เห็นถึงความสัมพันธ์ของ เวลาเปิด ปิด Interface network ของ Raspberry Pi4 กับขนาดไฟล์ที่ใช้ในการแบ่งข้อมูลผลการวิเคราะห์เพื่อใช้เป็นแนวทางในการหาเวลาเปิดปิด Interface network ที่เหมาะสมกับขนาดไฟล์ที่จะแบ่งที่ทำเวลาถ่ายโอนข้อมูลเฉลี่ยดีที่สุด

เมื่อนำผลเวลาเฉลี่ยการถ่ายโอนข้อมูลมาพล็อตกราฟแยกตามเวลาการเปิดปิด Interface network ของ Raspberry Pi4 เพื่อดูความสัมพันธ์ขนาดไฟล์กับเวลาในการโอนถ่ายข้อมูลสอดคล้องกับการตั้งเวลาเปิดปิดที่ค่าเวลาได้ดีที่สุดดังภาพ 36



ภาพ 36 แผนภูมิแสดงเวลาการโอนถ่ายข้อมูลเฉลี่ยเทียบกับขนาดการแบ่งไฟล์แยกตามเวลาเปิดปิด Interface ของ Raspberry Pi4



ภาพ 37 แผนภูมิแสดงเวลาการโอนถ่ายข้อมูลเฉลี่ยเทียบกับขนาดการแบ่งไฟล์แยกตามเวลาเปิดปิด Interface ของ Raspberry Pi4

จากภาพ 36, 37 แผนภูมิแสดงเวลารวมการโอนถ่ายข้อมูลเฉลี่ยเทียบกับขนาดการแบ่งไฟล์แยกตามเวลาเปิดปิด Interface ของ Raspberry Pi4 สรุปได้ว่าการแบ่งไฟล์ 1000Mb ที่ขนาดไฟล์ 500 – 560Mb เป็นช่วงการแบ่งขนาดไฟล์ที่ทำเวลาในการโอนถ่ายข้อมูลเฉลี่ยรวมได้ดีแต่ในระบบถ่ายโอนข้อมูลแบบกึ่งปิดผู้วิจัยจะทำการแบ่งไฟล์ข้อมูลด้วยโปรแกรม 7zip ที่ขนาดไฟล์ 550Mb ที่ทำเวลาการถ่ายโอนข้อมูลได้ดีที่สุดที่ 62.59 วินาที และเปิดปิด Interface network ของ Raspberry Pi4 ที่ 60 วินาที

2. ผลการทดลองจำลองสถานะการโดยการ Scan port และยิง Packet เข้าระบบและดูว่าโปรแกรม Snort สามารถตรวจจับ Packets ได้ที่เปอร์เซ็นต์และระบบสามารถจัดเก็บ log packets ที่ Firewall ตรวจสอบและ Block ได้ที่เปอร์เซ็นต์

การทดลองนี้มีจุดมุ่งหมายแสดงให้เห็นถึงประสิทธิภาพด้านความปลอดภัยของระบบ การทดลองทดสอบการโจมตีผู้วิจัยใช้โปรแกรม hping3 เป็นเครื่องมือในการทดสอบในการยิง Packet และ Port scan โจมตีระบบจากภายนอก โดยจะโจมตีทาง Interface Network ของ Raspberry Pi4 ที่เชื่อมต่อกับ Network ภายนอก ในการทดสอบ Port scan ได้ทำ TCP port scan ทดสอบสามครั้ง ครั้งละ 10 IP address scan port ของ IP ปลายทางของ Raspberry Pi4 แล้วดูว่าโปรแกรม Snort สามารถตรวจจับการโจมตีได้หรือไม่ ผลดังตารางที่ 5

ตาราง 5 ผลการทดสอบ Port scan ของ Raspberry Pi4

Source IP	Snort detect	Snort detect	Snort detect
192.168.1.2	YES	YES	YES
192.168.1.3	YES	YES	YES
192.168.1.4	YES	YES	YES
192.168.1.5	YES	YES	YES
192.168.1.6	YES	YES	YES
192.168.1.7	YES	YES	YES
192.168.1.8	YES	YES	YES
192.168.1.9	YES	YES	YES
192.168.1.10	YES	YES	YES
192.168.1.11	YES	YES	YES

จากตารางที่ 5 สรุปได้ว่า โปรแกรม Snort สามารถตรวจจับการโจมตีแบบ Port scan ได้ 100% และ ได้ทำการทดสอบ Scan port 22 ของ Raspberry PI4 ด้วยโปรโตคอล TCP ดังภาพ 38

```

root@nu-VirtualBox:~# hping3 --scan 22 -S 10.31.10.86
Scanning 10.31.10.86 (10.31.10.86), port 22
1 ports to scan, use -V to see all the replies
+-----+-----+-----+-----+-----+-----+-----+
|port| serv name | flags | ttl| id | win | len |
+-----+-----+-----+-----+-----+-----+
All replies received. Done.
Not responding ports: (22 ssh)
root@nu-VirtualBox:~#

```

ภาพ 38 ทดสอบ Scan port 22

จากภาพ 38 จะเห็นว่า port 22 ไม่มีการตอบสนองเนื่องจากระบบได้ใช้ Firewall block แต่ โปรแกรม Snort ก็ยังสามารถตรวจจับและแจ้งเตือนได้ดังภาพ 39 สาเหตุที่ Snort สามารถตรวจจับและแจ้งเตือนได้เพราะ Snort ทำงานใน Layer 2 Data link ซึ่งสามารถแยกประเภทโปรโตคอลต่างๆในระดับ Layer นี้ ส่วน Application firewalls จะตรวจสอบข้อมูลในระดับ Layer 3, 4, 5 และ 7 Network, Transport, Session และ Application ของ OSI Model

```

250:1900
03/26-22:33:14.675374  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] (UDP) 10.31.10.15:64757 -> 239.255.255.255
250:1900
03/26-22:33:18.690386  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] (UDP) 10.31.10.81:59474 -> 239.255.255.255
250:1900
03/26-22:33:18.742819  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] (UDP) 10.31.10.81:59475 -> 239.255.255.255
250:1900
03/26-22:33:18.903421  [**] [1:10000005:2] NMAP TCP Scan Port 22 [**] [Priority: 0] (TCP) 10.31.10.43:24509 -> 10.31.10.86:22
03/26-22:33:19.546476  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially Bad Traffic] [Priority: 2] (IGMP) 0.0.0.0 -> 224.0.0.1
03/26-22:33:19.697472  [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] (UDP) 10.31.10.81:59474 -> 239.255.255.255
250:1900

```

ภาพ 39 Snort detect scan port 22

การทดสอบยิง Packet เข้าระบบด้วยคำสั่ง hping3 -1 10.31.10.86 -c 1000 เป็นการ ping ส่ง Packet จำนวน 1000 packet ดังภาพ 40 โปรแกรม Snort สามารถตรวจจับได้ 100% ดังภาพ 41 และ 42

```

root@nu-VirtualBox: ~
len=46 ip=10.31.10.86 ttl=63 id=6281 icmp_seq=981 rtt=7.7 ms
len=46 ip=10.31.10.86 ttl=63 id=6282 icmp_seq=982 rtt=6.4 ms
len=46 ip=10.31.10.86 ttl=63 id=6283 icmp_seq=983 rtt=6.2 ms
len=46 ip=10.31.10.86 ttl=63 id=6284 icmp_seq=984 rtt=1.9 ms
len=46 ip=10.31.10.86 ttl=63 id=6285 icmp_seq=985 rtt=9.7 ms
len=46 ip=10.31.10.86 ttl=63 id=6286 icmp_seq=986 rtt=9.3 ms
len=46 ip=10.31.10.86 ttl=63 id=6287 icmp_seq=987 rtt=8.9 ms
len=46 ip=10.31.10.86 ttl=63 id=6288 icmp_seq=988 rtt=8.9 ms
len=46 ip=10.31.10.86 ttl=63 id=6289 icmp_seq=989 rtt=7.8 ms
len=46 ip=10.31.10.86 ttl=63 id=6290 icmp_seq=990 rtt=7.6 ms
len=46 ip=10.31.10.86 ttl=63 id=6291 icmp_seq=991 rtt=7.3 ms
len=46 ip=10.31.10.86 ttl=63 id=6292 icmp_seq=992 rtt=3.7 ms
len=46 ip=10.31.10.86 ttl=63 id=6293 icmp_seq=993 rtt=2.4 ms
len=46 ip=10.31.10.86 ttl=63 id=6294 icmp_seq=994 rtt=2.2 ms
len=46 ip=10.31.10.86 ttl=63 id=6295 icmp_seq=995 rtt=1.7 ms
len=46 ip=10.31.10.86 ttl=63 id=6296 icmp_seq=996 rtt=9.6 ms
len=46 ip=10.31.10.86 ttl=63 id=6297 icmp_seq=997 rtt=9.2 ms
len=46 ip=10.31.10.86 ttl=63 id=6298 icmp_seq=998 rtt=8.8 ms
len=46 ip=10.31.10.86 ttl=63 id=6299 icmp_seq=999 rtt=7.9 ms

--- 10.31.10.86 hping statistic ---
1000 packets transmitted, 1000 packets received, 0% packet loss
round-trip min/avg/max = 1.4/5.9/19.8 ms
root@nu-VirtualBox:~#

```

၈၇၇ 40 Ping packets 1000 packet

```

pi@raspberrypi: /etc/...
File Edit Tabs Help
250:1900
03/26-22:52:57.349608 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [ICMP] 10.31.10.43 -> 10.31.10.86
03/26-22:52:57.349608 [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] [ICMP] 10.31.10.43 -> 10.31.10.86
03/26-22:52:57.349636 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [ICMP] 10.31.10.86 -> 10.31.10.43
03/26-22:52:58.176296 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] [UDP] 10.31.10.36:63262 -> 239.255.255.250:1900
03/26-22:52:58.350952 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [ICMP] 10.31.10.43 -> 10.31.10.86
03/26-22:52:58.350952 [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] [ICMP] 10.31.10.43 -> 10.31.10.86
03/26-22:52:58.350968 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [ICMP] 10.31.10.86 -> 10.31.10.43
03/26-22:52:59.177068 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] [UDP] 10.31.10.36:60127 -> 239.255.255.250:1900
03/26-22:52:59.292511 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [IPv6-ICMP] fe80::5ee9:31ff:feb5:6ccc -> ff02::2
03/26-22:52:59.350870 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [ICMP] 10.31.10.43 -> 10.31.10.86
03/26-22:52:59.350870 [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] [ICMP] 10.31.10.43 -> 10.31.10.86
03/26-22:52:59.350955 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [ICMP] 10.31.10.86 -> 10.31.10.43
03/26-22:52:59.597754 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] [UDP] 10.31.10.23:59213 -> 239.255.255.250:1900
03/26-22:53:00.351278 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [ICMP] 10.31.10.43 -> 10.31.10.86
03/26-22:53:00.351278 [**] [1:469:3] ICMP PING NMAP [**] [Classification: Attempted Information Leak] [Priority: 2] [ICMP] 10.31.10.43 -> 10.31.10.86
03/26-22:53:00.351387 [**] [1:10000001:1] ICMP ping scan [**] [Priority: 0] [ICMP] 10.31.10.86 -> 10.31.10.43
03/26-22:53:00.670832 [**] [1:1917:6] SCAN UPnP service discover attempt [**] [Classification: Detection of a Network Scan] [Priority: 3] [UDP] 10.31.10.23:59213 -> 239.255.255.250:1900

```

၈၇၇ 41 Snort detect ICMP

```

pi@raspberrypi: /etc/... 2024-03-26-223324... Pictures [pi]
pi@raspberrypi: /etc/snort

WARNING: No preprocessors configured for policy 0.
03/26-23:08:23.797998 10.31.10.43 -> 10.31.10.86
ICMP TTL:63 TOS:0x0 ID:30977 IPlen:20 DgLen:28
Type:8 Code:0 ID:1 Seq:1151 ECHO
-----
WARNING: No preprocessors configured for policy 0.
03/26-23:08:23.797998 10.31.10.43 -> 10.31.10.86
ICMP TTL:63 TOS:0x0 ID:30977 IPlen:20 DgLen:28
Type:8 Code:0 ID:1 Seq:1151 ECHO
-----
WARNING: No preprocessors configured for policy 0.
03/26-23:08:23.798149 10.31.10.86 -> 10.31.10.43
ICMP TTL:64 TOS:0x0 ID:35932 IPlen:20 DgLen:28
Type:0 Code:0 ID:1 Seq:1151 ECHO REPLY
-----
WARNING: No preprocessors configured for policy 0.
03/26-23:08:23.880869 10.31.10.20:49792 -> 239.255.255.250:1900
UDP TTL:1 TOS:0x0 ID:4579 IPlen:20 DgLen:202
Len: 174
-----
WARNING: No preprocessors configured for policy 0.
03/26-23:08:24.086539 10.31.10.36:43486 -> 239.255.255.250:1900
UDP TTL:1 TOS:0x0 ID:45509 IPlen:20 DgLen:202
Len: 174
-----
WARNING: No preprocessors configured for policy 0.
03/26-23:08:24.798186 10.31.10.43 -> 10.31.10.86
ICMP TTL:63 TOS:0x0 ID:30978 IPlen:20 DgLen:28
Type:8 Code:0 ID:1 Seq:1152 ECHO
-----
WARNING: No preprocessors configured for policy 0.
03/26-23:08:24.798186 10.31.10.43 -> 10.31.10.86
ICMP TTL:63 TOS:0x0 ID:30978 IPlen:20 DgLen:28
Type:8 Code:0 ID:1 Seq:1152 ECHO
-----
WARNING: No preprocessors configured for policy 0.
03/26-23:08:24.798264 10.31.10.86 -> 10.31.10.43
ICMP TTL:64 TOS:0x0 ID:36178 IPlen:20 DgLen:28
Type:0 Code:0 ID:1 Seq:1152 ECHO REPLY
-----
WARNING: No preprocessors configured for policy 0.

```

```

Breakdown by protocol (includes rebuilt packets):
Eth:      6214 (100.000%)
VLAN:     0 ( 0.000%)
IP4:      5703 ( 91.777%)
Frag:     0 ( 0.000%)
ICMP:     3000 ( 48.278%)
UDP:      2693 ( 43.338%)
TCP:      0 ( 0.000%)

```

ภาพ 42 Snort log detect ICMP

จากภาพ 42 พบว่า โปรแกรม Snort จับเก็บ log ping 1 packet จะเก็บ 3 record คือ Type 8 ECHO 2 record และ Type 0 ECHO REPLY 1 record จากการสรุปภาพรวมพบว่า Snort ตรวจเจอ ICMP 3000 ครั้ง เมื่อนำมาหารด้วยจำนวน record ของ 1 Packet ก็จะได้จำนวน Packet ที่ส่งมาคือ 1000 packet

ทดสอบ Scan port 22 ของ Raspberry Pi4 ด้วยคำสั่ง hping3 --scan 22 -S 10.31.10.86 10 ครั้งแล้วตรวจสอบ log ของ firewall พบว่าทุกครั้งที่ทำกร Scan port firewall สามารถ Block และเก็บ log ไว้ได้ทุกครั้ง คิดเป็น 100% ดังภาพ 43


```

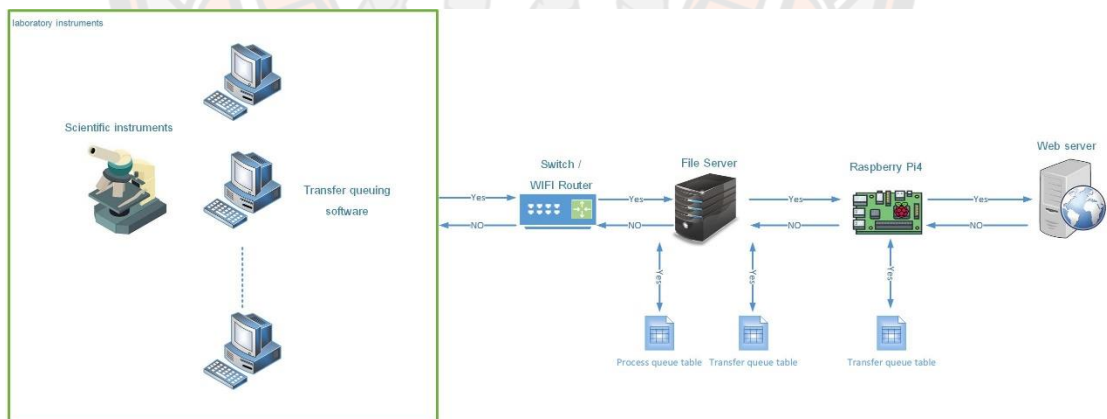
File Edit Tabs Help
pi@raspberrypi: /etc/... 2024-03-26-223324... Pictures [pi]
pi@raspberrypi: /etc/snort

1 ID=0 DF PROTO=2
Mar 27 00:25:01 raspberrypi kernel: [23719.238591] [UFW BLOCK] IN=eth1 OUT= MAC=01:00:5e:00:00:fb:e4:a8:df:8d:5c:9a:08:00 SRC=10.31.10.40 DST=224.0.0.251 LEN=32 TOS=0x00 PREC=0x00
0 TTL=1 ID=29052 PROTO=2
Mar 27 00:27:06 raspberrypi kernel: [23844.426744] [UFW BLOCK] IN=eth1 OUT= MAC=01:00:5e:00:00:fb:e4:a8:df:8d:5c:9a:08:00 SRC=0.0.0.0 DST=224.0.0.1 LEN=32 TOS=0x00 PREC=0x00 TTL=1
1 ID=0 DF PROTO=2
Mar 27 00:27:06 raspberrypi kernel: [23844.426744] [UFW BLOCK] IN=eth1 OUT= MAC=01:00:5e:00:00:fb:e4:a8:df:8d:5c:9a:08:00 SRC=10.31.10.28 DST=224.0.0.251 LEN=32 TOS=0x00 PREC=0x00
0 TTL=1 ID=53410 PROTO=2
Mar 27 00:28:15 raspberrypi kernel: [23913.074946] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31026 DF PROTO=TCP SPT=28428 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:28:16 raspberrypi kernel: [23914.074411] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31027 DF PROTO=TCP SPT=28428 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:28:18 raspberrypi kernel: [23916.075146] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31028 DF PROTO=TCP SPT=28428 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:28:22 raspberrypi kernel: [23920.074925] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31029 DF PROTO=TCP SPT=28428 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:28:30 raspberrypi kernel: [23928.074324] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31030 DF PROTO=TCP SPT=28428 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:29:05 raspberrypi kernel: [23983.272066] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31031 DF PROTO=TCP SPT=28453 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:29:06 raspberrypi kernel: [23984.271634] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31032 DF PROTO=TCP SPT=28453 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:29:08 raspberrypi kernel: [23986.272576] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31033 DF PROTO=TCP SPT=28453 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:29:11 raspberrypi kernel: [23969.868459] [UFW BLOCK] IN=eth1 OUT= MAC=01:00:5e:00:00:01:5c:e9:31:b5:6c:cc:08:00 SRC=0.0.0.0 DST=224.0.0.1 LEN=32 TOS=0x00 PREC=0x00 TTL=1
1 ID=0 DF PROTO=2
Mar 27 00:29:11 raspberrypi kernel: [23969.893157] [UFW BLOCK] IN=eth1 OUT= MAC=01:00:5e:00:00:fb:98:ee:cb:c2:bd:44:08:00 SRC=10.31.10.37 DST=224.0.0.251 LEN=32 TOS=0x00 PREC=0x00
0 TTL=1 ID=54075 PROTO=2
Mar 27 00:29:12 raspberrypi kernel: [23970.274196] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31034 DF PROTO=TCP SPT=28453 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:29:20 raspberrypi kernel: [23978.274634] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31035 DF PROTO=TCP SPT=28505 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:29:46 raspberrypi kernel: [24004.473605] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31036 DF PROTO=TCP SPT=28505 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:29:47 raspberrypi kernel: [24005.474346] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31037 DF PROTO=TCP SPT=28505 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:30:01 raspberrypi kernel: [24019.477359] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31040 DF PROTO=TCP SPT=28505 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:30:16 raspberrypi kernel: [24034.043898] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31043 DF PROTO=TCP SPT=28519 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:30:36 raspberrypi kernel: [24054.432993] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31046 DF PROTO=TCP SPT=28532 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:30:56 raspberrypi kernel: [24073.994481] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31051 DF PROTO=TCP SPT=28542 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:31:16 raspberrypi kernel: [24094.407838] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31058 DF PROTO=TCP SPT=28556 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:31:36 raspberrypi kernel: [24113.969192] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31065 DF PROTO=TCP SPT=28564 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
Mar 27 00:31:56 raspberrypi kernel: [24134.824766] [UFW BLOCK] IN=eth1 OUT= MAC=00:e0:4c:68:05:37:fd:de:f1:47:22:25:08:00 SRC=10.31.10.43 DST=10.31.10.86 LEN=52 TOS=0x00 PREC=0x00
0 TTL=128 ID=31075 DF PROTO=TCP SPT=28575 DPT=22 WINDOW=64240 RES=0x00 SYN URG=0
root@raspberrypi:~# cat /var/log/

```

ภาพ 43 แสดง log firewall block scan port 22

2. ผลการทดลองความถูกต้องของข้อมูลผลการวิเคราะห์ที่ได้รับผ่านทาง website



ภาพ 44 แสดงลำดับการถ่ายโอนและการตรวจสอบความถูกต้องของข้อมูล

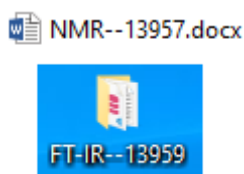
ในการทดลองนี้ผู้วิจัยได้ทำการทดลองโอนถ่ายข้อมูลผ่านระบบด้วยไฟล์ข้อมูลขนาดสูงสุดที่ 1000MB โดยมีการแบ่งไฟล์ขนาด 1000MB เป็นไฟล์ขนาดต่าง ๆ ตั้งแต่ 10MB ถึง 1000MB ถ่ายโอนที่เวลาเปิด ปิด Interface network ของ Raspberry Pi4 ที่ 55, 60, 65, 70 และ 75 วินาที ผลการทดลองพบว่าการถ่ายโอนข้อมูลใช้เวลาเปิด ปิด Interface network ของ Raspberry Pi4 ที่ 55, 60, 65, 70 และ 75 วินาทีระบบสามารถโอนถ่ายไฟล์ข้อมูลได้ถูกต้องสูงสุดที่ 760MB, 860MB ,

910MB, 1000MB, 1000MB ตามลำดับ ในขั้นตอนการถ่ายโอนข้อมูลระบบมีการตรวจสอบความถูกต้องของข้อมูลดังภาพ 44 ตั้งแต่การใช้งาน Transfer queuing software ถ่ายโอนข้อมูลจากเครื่องต้นทางไปเก็บไว้ยัง File server โดยตรวจสอบไฟล์หรือไดเรกทอรีที่โอนถ่ายสำเร็จ (Yes) จะถูกเขียนคิวลงฐานข้อมูลตาราง Process queue ของ File server ถ้าไม่สำเร็จ (No) จะมีการแจ้งเตือนผู้ใช้งาน การตรวจสอบความถูกต้องของข้อมูลจาก File server โอนถ่ายข้อมูลไปยัง Raspberry Pi4 จะเริ่มจากขั้นตอนการแบ่งไฟล์และเข้ารหัสลับข้อมูลแบบ AES มีการหาจำนวนไฟล์ที่แบ่งได้และขนาดไฟล์ที่แบ่ง เก็บในฐานข้อมูลตาราง Transfer queue มีรูปแบบจัดเก็บเป็น จำนวนไฟล์ที่แบ่งจากโปรแกรม7zip@ขนาดไฟล์ที่แบ่งได้@ชื่อไฟล์.7z.xxx.aes การตรวจสอบความถูกต้องจะเริ่มจากอ่านคิวจากตาราง Transfer queue แล้วโอนถ่ายไฟล์ไปยัง Raspberry Pi4 ถ้าไฟล์ไหนโอนถ่ายสำเร็จ (Yes) ให้ลบคิวไฟล์นั้นออกจากตาราง Transfer queue ของ File server และเขียนคิวในตาราง Transfer queue ของ Raspberry Pi4 โดยใช้วิธีตรวจสอบจากขนาดไฟล์ที่โอนถ่ายไปมีขนาดเท่ากับขนาดไฟล์ที่แบ่งได้จากโปรแกรม 7ZIP ที่เก็บในฐานข้อมูลตาราง Transfer queue และถ้าไฟล์ไหนโอนถ่ายไม่สำเร็จ (No) ก็ให้ทำการโอนถ่ายใหม่หรือถ้า Connection ตัดก็ให้โอนถ่ายในรอบต่อไปจนกว่าคิวในตาราง Transfer queue ของ File server จะถูกลบ การตรวจสอบความถูกต้องของการโอนถ่ายไฟล์จาก Raspberry Pi4 ไปยัง Web server จะเริ่มจากการอ่านข้อมูลจากตาราง Transfer queue และโอนถ่ายข้อมูลตามคิวไป Web server ถ้าไฟล์ไหนโอนถ่ายสำเร็จ (Yes) ให้ลบคิวไฟล์นั้นออกจากตาราง Transfer queue ของ Raspberry Pi4 และเขียนคิวใน Web server ถ้าไฟล์ไหนโอนถ่ายไม่สำเร็จ (No) ก็ให้ทำการโอนถ่ายใหม่หรือถ้า Connection ตัดก็ให้โอนถ่ายในรอบต่อไป Web server จะทำการตรวจสอบคิวถ้ามีคิวและมีไฟล์ระบบจะทำการถอดรหัสลับแบบ AES เก็บไฟล์ที่ถอดรหัสสำเร็จไว้ในไดเรกทอรี 7ZIP_FILE ในขณะเดียวกัน Web server จะทำการตรวจสอบว่าไฟล์ที่อยู่ใน 7ZIP_FILE มีจำนวนครบตามการแบ่งหรือไม่ถ้าครบก็จะทำการรวมไฟล์และให้บริการผ่านทางเว็บไซต์ ข้อดีของการใช้โปรแกรม 7ZIP ในระบบคือโปรแกรมจะช่วยให้การตรวจสอบความถูกต้องสมบูรณ์ของไฟล์เพื่อให้ได้ไฟล์ข้อมูลครบถ้วนพร้อมที่จะทำการรวมไฟล์ ถ้าไฟล์ไม่สมบูรณ์หรือมีจำนวนไม่ครบโปรแกรมจะไม่สามารถรวมไฟล์และให้บริการได้ ดังนั้นสามารถสรุปได้ว่าไฟล์ข้อมูลที่ได้จากระบบมีความสมบูรณ์พร้อมให้บริการผ่านทางเว็บไซต์ได้ถูกต้อง

2.1 การใช้งานระบบถ่ายโอนข้อมูลแบบกึ่งปิดมีขั้นตอนดังนี้

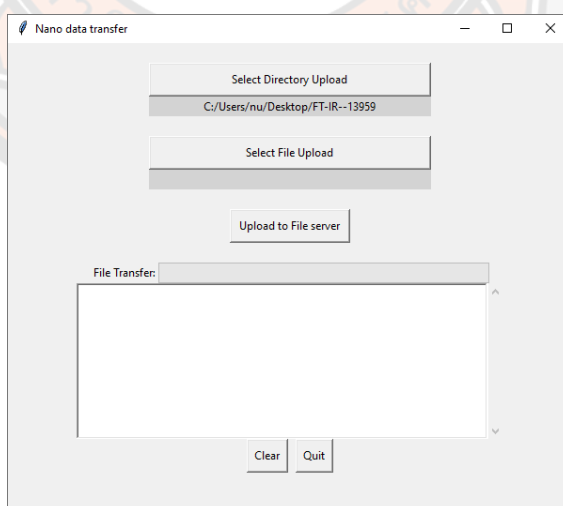
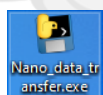
2.1.1 ในการใช้งานระบบถ่ายโอนข้อมูลแบบกึ่งปิดจำเป็นต้องมีการตั้งชื่อไฟล์และ Folders ให้เชื่อมโยงไปยังเลขที่จองเพื่อให้ข้อมูลและเลขที่จองมีการเชื่อมโยงกัน รูปแบบการตั้งชื่อไฟล์หรือ Folders มีรูปแบบดังนี้ ชื่อเครื่องมือ--เลขที่จอง เช่น ตั้งชื่อ Folders เก็บไฟล์ข้อมูลผลการวิเคราะห์

ของเครื่อง FT-IR เลขที่จองที่ 13959 จะได้เป็น FT-IR--13959 และตั้งชื่อไฟล์ผลการวิเคราะห์เครื่อง NMR เลขที่จอง 13957 จะได้เป็น NMR--13957.docx ดังภาพ 45

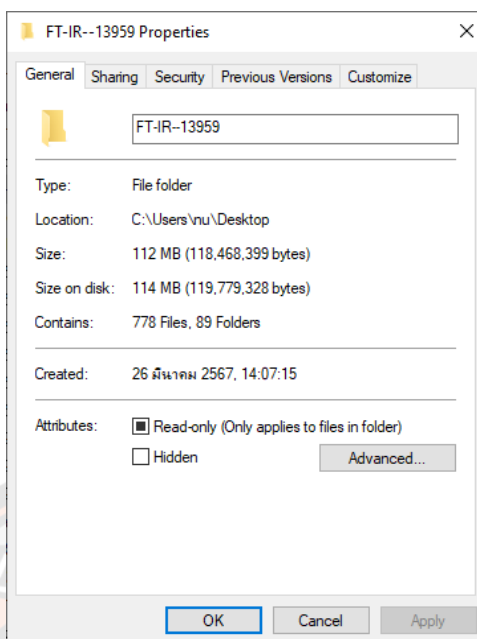


ภาพ 45 แสดงตัวอย่างการตั้งชื่อไฟล์ และ Folders

2.1.2 เปิดโปรแกรม Nano data transfer ที่เป็น Transfer queuing software ที่เครื่องควบคุมการทำงานเครื่องมือวิทยาศาสตร์ดังภาพ 46 เพื่อทำการส่งไฟล์หรือ Folders ไปยัง File server ทำการประมวลผลส่งต่อไปยัง Raspberry Pi4 และ Raspberry Pi4 ส่งต่อไปยัง Web server เช่นต้องการส่งไฟล์ข้อมูลผลการวิเคราะห์ของเครื่อง FT-IR เลขที่จองที่ 13959 ให้คลิกที่ Select Directory Upload แล้วเลือก Folders FT-IR--13959 แล้วคลิกที่ Upload to File server แล้วรอให้ระบบทำการประมวลผลและส่งต่อข้อมูลไปยังเครื่อง Web server



ภาพ 46 แสดงโปรแกรม Nano data transfer



ภาพ 47 แสดงขนาด Folders จำนวนไฟล์และจำนวน Folders ก่อนส่งไปยัง File server

2.1.3 เข้าเว็บไซต์ตรวจสอบข้อมูลผลการวิเคราะห์ที่ส่งผ่านระบบและดาวโหลดมาใช้งานใน
เว็บไซต์ให้บริการข้อมูลจะแยกประเภทผู้ใช้งานออกเป็นสองประเภทคือ Admin และ Clients โดย
Admin คือนักวิทยาศาสตร์ที่ดูแลเครื่องมือวิทยาศาสตร์ และ Clients คือผู้รับบริการ เมื่อ Admin
หรือนักวิทยาศาสตร์เข้าสู่ระบบในหน้าเว็บไซต์จะเห็นรายการจองเครื่องมือวิทยาศาสตร์ที่ดูแลเท่านั้น
ดังภาพ 48

รายการใช้งานเครื่องมือทั้งหมด

ประจำเดือน: 03/01/2024 ถึงเดือน: 03/28/2024

แสดงทั้งหมด 40 รายการ

#	เลขที่จอง	ชื่อ-นามสกุล	เครื่องมือที่ใช้	วันที่วิเคราะห์	รายได้ที่ได้รับจริงและคิดส่วนลดแล้ว(บาท)	สถานะการจ่ายเงิน	ผลการวิเคราะห์
1	slcs-13981	ธนัชชา แสงทอง	FT-IR	2024-03-27	250	●	■
2	slcs-13977	อนุสรณ์ แสนสี	NMR	2024-03-27	525	●	■
3	slcs-13971	ธนัชชา แสงทอง	FT-IR	2024-03-25	0	○	■
4	slcs-13962	อนุสรณ์ แสนสี	NMR	2024-03-26	175	●	■
5	slcs-13959	วงศ์กร เหล่าวิไลด์	FT-IR	2024-03-25	250	●	■
6	slcs-13957	รัตนาภรณ์ จันทกุล	NMR	2024-03-25	250	●	■

ภาพ 48 แสดงรายการจองเครื่องมือวิทยาศาสตร์ทั้งหมดที่นักวิทยาศาสตร์รับผิดชอบดูแล

จากภาพ 48 คอลัมน์ผลการวิเคราะห์จะเห็นไอคอนรูปกระดาษเป็นสีแดงแสดงว่ายังไม่มีข้อมูลผลการวิเคราะห์ ถ้าไอคอนรูปกระดาษเป็นสีเขียวแสดงว่ามีข้อมูลผลการวิเคราะห์สามารถดาวน์โหลดข้อมูลได้

ดังภาพ 49

รายการใช้งานเครื่องมือทั้งหมด

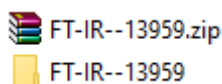
ประจำเดือน: 03/01/2024 ถึงเดือน: 03/28/2024

แสดงทั้งหมด 40 รายการ

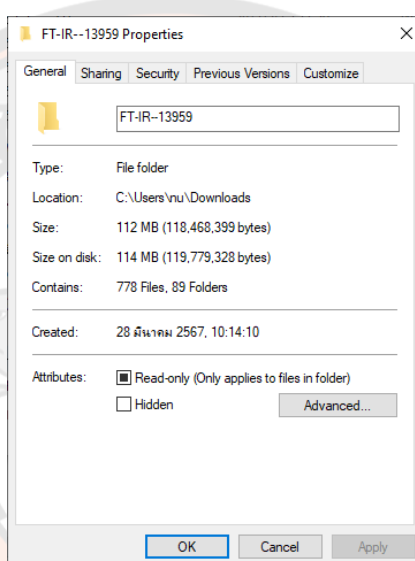
#	เลขที่จอง	ชื่อ-นามสกุล	เครื่องมือที่ใช้	วันที่วิเคราะห์	รายได้ที่ได้รับจริงและคิดส่วนลดแล้ว(บาท)	สถานะการจ่ายเงิน	ผลการวิเคราะห์
1	slcs-13981	ธนัชชา แสงทอง	FT-IR	2024-03-27	250	●	■
2	slcs-13977	อนุสรณ์ แสนสี	NMR	2024-03-27	525	●	■
3	slcs-13971	ธนัชชา แสงทอง	FT-IR	2024-03-25	0	○	■
4	slcs-13962	อนุสรณ์ แสนสี	NMR	2024-03-26	175	●	■
5	slcs-13959	วงศ์กร เหล่าวิไลด์	FT-IR	2024-03-25	250	●	■
6	slcs-13957	รัตนาภรณ์ จันทกุล	NMR	2024-03-25	250	●	■

ภาพ 49 แสดงรายการจองเครื่องมือวิทยาศาสตร์ทั้งหมดที่นักวิทยาศาสตร์รับผิดชอบดูแล
ที่มีข้อมูลผลการวิเคราะห์

จากภาพ 49 จะเห็นว่ารายการจองที่ slcs-13959 มีข้อมูลผลการวิเคราะห์ที่ได้ทำการ Upload เข้าระบบในหัวข้อ 2.1.2 ถ้านักวิทยาศาสตร์ต้องการนำผลการวิเคราะห์ส่งให้ผู้รับบริการให้คลิกที่ไอคอนกระดาษสีเขียว เมื่อคลิกระบบจะทำการสร้างไฟล์ zip และดาวน์โหลดอัตโนมัติดังภาพ 50



ภาพ 50 แสดงไฟล์ที่โหลดจากระบบ และผลจากการ unzip ไฟล์



ภาพ 51 แสดงขนาด Folders จำนวนไฟล์และจำนวน Folders ที่ได้จากระบบ

จากภาพ 47 และ 51 จะพบว่าข้อมูลที่ได้จากระบบมีขนาดและจำนวนไฟล์และ Folders เท่ากันกับข้อมูลก่อนนำเข้าสู่ระบบ ส่วน clients หรือผู้รับบริการเมื่อเข้าสู่ระบบก็จะเห็นรายการจองของตนเอง และสามารถดาวน์โหลดผลการวิเคราะห์ผ่านเว็บไซต์ได้เหมือนนักวิทยาศาสตร์ ถ้ารายการจองนั้นไอคอนกระดาษเป็นสีเขียวดังภาพ 52

แสดงทั้งหมด 27 รายการ

#	เลขที่จอง	ชื่อ-นามสกุล	เครื่องที่ใช้	วันที่วิเคราะห์	รายได้ที่ได้รับจริง และคิดส่วนลด แล้ว(บาท)	สถานะการจ่ายเงิน	ผลการวิเคราะห์
1	slcs-8831	รัตนาภรณ์ ฉันทกุล	NMR	2021-09-16	850	●	■
2	slcs-8865	รัตนาภรณ์ ฉันทกุล	NMR	2021-09-22	1150	●	■
3	slcs-10739	รัตนาภรณ์ ฉันทกุล	NMR	2022-08-22	2450	●	■
4	slcs-10738	รัตนาภรณ์ ฉันทกุล	NMR	2022-08-22	0	○	■
5	slcs-11257	รัตนาภรณ์ ฉันทกุล	NMR	2022-10-26	250	●	■
6	slcs-13935	รัตนาภรณ์ ฉันทกุล	NMR	2024-03-21	250	●	■
7	slcs-13957	รัตนาภรณ์ ฉันทกุล	NMR	2024-03-25	250	●	■

ภาพ 52 แสดงรายการจองเครื่องมือวิทยาศาสตร์ทั้งหมดของผู้รับบริการที่มีข้อมูลผลการวิเคราะห์



บทที่ 5

สรุปผลการวิจัย อภิปรายผล และข้อเสนอแนะ

การวิจัยเรื่อง ระบบถ่ายโอนข้อมูลแบบกึ่งปิด มีวัตถุประสงค์เพื่อ 1) ได้ระบบถ่ายโอนข้อมูลแบบกึ่งปิดที่นำข้อมูลขึ้นบนอินเทอร์เน็ตได้ 2) ระบบจะหลีกเลี่ยงการเพิ่มช่องโหว่ทางด้านความปลอดภัยต่อเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิเคราะห์ และข้อมูลผลการวิเคราะห์ 3) ระบบให้บริการข้อมูลผลการวิเคราะห์ผ่านทาง website การวิจัยครั้งนี้เป็นการพัฒนาระบบขึ้นมาเพื่ออำนวยความสะดวกให้กับนักวิทยาศาสตร์และผู้รับบริการของศูนย์เครื่องมือวิทยาศาสตร์ให้สามารถนำผลการวิเคราะห์ตัวอย่างออกจากเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิทยาศาสตร์มาใช้งานได้อย่างถูกต้องและปลอดภัยผ่านระบบเครือข่าย network แทนการใช้ CD หรือ DVD เนื่องจากทางศูนย์เครื่องมือวิทยาศาสตร์มีนโยบายห้ามนำอุปกรณ์ภายนอกมาเชื่อมต่อกับเครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิทยาศาสตร์เช่น Flash drive, External hard drive เพื่อป้องกันไม่ให้เครื่องคอมพิวเตอร์ควบคุมการทำงานของเครื่องมือวิทยาศาสตร์ถูกโจมตีด้วย Malware , ไวรัสคอมพิวเตอร์ และจากผู้ไม่ประสงค์ดี

1.สรุปผลการวิจัย

การวิจัยครั้งนี้ได้ระบบถ่ายโอนข้อมูลแบบกึ่งปิดที่มีการแบ่งขนาดไฟล์ จำนวนไฟล์และเวลาเปิดปิด Interface network สัมพันธ์กับเวลาในการโอนถ่ายข้อมูลที่มีค่า R square เท่ากับ 0.7684 หรือเท่ากับร้อยละ 76.84 โดยมีช่วงการแบ่งไฟล์ที่ 500-560MB และเวลาเปิดปิด Interface network ของ Raspberry Pi4 ที่ 60 วินาที ระบบได้มีการตั้งค่าการแบ่งไฟล์ที่ 550MB ซึ่งใช้เวลาเฉลี่ยโอนถ่ายข้อมูลดีที่สุด ระบบที่พัฒนาขึ้นจะประกอบไปด้วยโปรแกรม Nano data transfer สำหรับส่งข้อมูลผลการวิเคราะห์จากเครื่องควบคุมการทำงานของเครื่องมือวิทยาศาสตร์ไปยัง File server เพื่อทำการประมวลผลและเข้ารหัสลับข้อมูลก่อนส่งต่อไปยังตัวกลางการโอนถ่ายข้อมูลคือ Raspberry Pi4 โดยโปรแกรม Nano data transfer จะถูกติดตั้งในเครื่องควบคุมการทำงานของเครื่องมือวิทยาศาสตร์ ส่วนใน File server จะมีโปรแกรมทำงานอยู่สองโปรแกรมคือ process_send_file และ send_fileto_raspberry โปรแกรม process_send_file จะทำหน้าที่นำไฟล์หรือไดเรกทอรีที่ได้จากโปรแกรม Nano data transfer บีบอัดและแบ่งไฟล์หรือไดเรกทอรี ออกเป็นไฟล์ละ 550MB ซึ่งเป็นขนาดไฟล์ที่ทำเวลาโอนถ่ายเฉลี่ยร่วมได้ดีที่สุดจากการทดลองข้างต้น พร้อมกับเข้ารหัสลับแบบ AES เพิ่มความปลอดภัยให้กับข้อมูล โปรแกรม send_fileto_raspberry จะทำหน้าที่ตรวจสอบ Connection กับ Raspberry Pi4 ถ้ามี Connection จะนำเอาไฟล์ข้อมูลที่เข้ารหัสลับแบบ AES ส่งไปยัง Raspberry Pi4 ตามคิว ส่วน Raspberry Pi4 ทำหน้าที่เป็นตัวกลางในการโอนถ่ายข้อมูลที่ถูก

เข้ารหัสลับแบบ AES แล้วส่งไปยังเครื่อง Web server โดยควบคุมการทำงานด้วยโปรแกรม send_fileto_fileservice ซึ่ง Raspberry Pi4 มีการทำงานประกอบไปด้วย

1) ทำหน้าที่ควบคุมการเปิดปิด interface network ทุก ๆ 60 วินาที เวลาที่จะมีความสัมพันธ์กับขนาดการแบ่งไฟล์ที่ 550MB ที่ทำเวลาโอนถ่ายข้อมูลเฉลี่ยรวมได้ดีที่สุดจากผลการทดลองข้างต้น

2) ทำหน้าที่เป็น Firewall โดยจะกำหนดให้สามารถรับ income packet จาก IP address ของเครื่อง File server เท่านั้น ส่วน interface network ที่เชื่อมต่อกับ network ภายนอกไปยังเครื่อง Web server จะกำหนดให้ Packet outcome อย่างเดียวและ Block income ทั้งหมด firewall สามารถจัดเก็บ log ได้ 100%

3) ทำหน้าที่เป็น network intrusion detection โดยใช้งานโปรแกรม Snort แจ้งเตือนเมื่อมีการพยายาม Scan port หรือยิง packet จากภายนอกเข้ามายังตัวของ Raspberry Pi4 ซึ่งระบบสามารถทำการตรวจสอบเจอ 100% ถึงแม้ firewall จะ block เพราะ Snort ทำงานใน Layer 2 Data link ซึ่งสามารถแยกประเภทโปรโตคอลต่าง ๆ ในระดับ Layer นี้ได้

4) ทำหน้าที่โอนถ่ายข้อมูลที่ถูกเข้ารหัสลับแบบ AES แล้วส่งไปยังเครื่อง Web server ตามคิวเมื่อมีการเชื่อมต่อกับ Web server และตัดการเชื่อมต่อกับ File server

เครื่อง Web server มีหน้าที่ถอดรหัสลับ AES และรวมไฟล์ด้วยโปรแกรม 7zip ตามคิวที่ได้รับจาก Raspberry Pi4 และทำหน้าที่เป็นเว็บแอปพลิเคชันให้บริการข้อมูลแก่นักวิทยาศาสตร์และผู้รับบริการ ข้อมูลที่ได้รับจากระบบมีความถูกต้อง 100% จากผลการทดลองข้างต้นที่ระบบมีการตรวจสอบความถูกต้องของข้อมูลระหว่างการถ่ายโอนและมีการนำโปรแกรม 7ZIP ใช้งานในระบบช่วยในการตรวจสอบความถูกต้องสมบูรณ์ของไฟล์เพื่อให้ได้ไฟล์ข้อมูลครบถ้วนพร้อมที่จะทำการรวมไฟล์ ถ้าไฟล์ไม่สมบูรณ์หรือมีจำนวนไม่ครบโปรแกรมจะไม่สามารถรวมไฟล์และให้บริการได้

2. อภิปรายผล

ระบบถ่ายโอนข้อมูลแบบกึ่งปิดที่ได้จากการวิจัยครั้งนี้ระบบสามารถทำงานได้ถูกต้องสามารถที่จะนำผลการวิเคราะห์ออกจากเครื่องควบคุมการทำงานของเครื่องมือวิทยาศาสตร์และให้บริการข้อมูลแก่นักวิทยาศาสตร์และผู้รับบริการได้แต่ต้องใช้เวลาในการส่งต่อและประมวลผลนานพอสมควร ทั้งนี้ขึ้นอยู่กับค่าเวลาการทำงานภายในของระบบเช่น ค่าการทำงานของโปรแกรมน้อย ๆ ในระบบที่มีการทำงานแบบวนซ้ำ ๆ เวลาในการเปิดปิด Interface network ขนาดไฟล์ จำนวนไฟล์ที่แบ่งได้ ประสิทธิภาพของเครื่อง File server, Raspberry Pi4, Web server และระบบ Network ซึ่งในการวิจัยและพัฒนาครั้งนี้เครื่องคอมพิวเตอร์ที่ใช้ทำเป็นเครื่อง File server และ Web server เป็นคอมพิวเตอร์ PC ธรรมดาไม่มีสเปคต่ำ จึงส่งผลต่อเวลาในการใช้งานระบบ ในการใช้งานระบบจำเป็นต้องมีกฎการตั้งชื่อเพื่อให้ไฟล์ข้อมูลผลการวิเคราะห์เชื่อมโยงกับเลขที่การจองในเว็บไซต์

สำหรับให้บริการข้อมูลผลการวิเคราะห์ การให้บริการผลการวิเคราะห์ผ่านทางเว็บไซต์เป็นการบริการภายในมหาวิทยาลัยเพราะผู้รับบริการส่วนใหญ่เป็นนิสิตและบุคลากร และเพื่อป้องกันไม่ให้ถูกโจมตีจากผู้ไม่ประสงค์ดีจากภายนอกมหาวิทยาลัย การใช้งาน Raspberry Pi4 ทำหน้าที่เป็น Firewall Network intrusion detection และ Database พร้อมกับรันโปรแกรม send_fileto_fileservice ที่คอยเปิด ปิด Interface network และส่งไฟล์ข้อมูลไปยัง Web server ทรัพยากรของ Raspberry Pi4 จะไม่พออาจจะส่งผลทำให้ระบบทำงานช้าลง

3. ข้อเสนอแนะ

ระบบถ่ายโอนข้อมูลแบบกึ่งปิดที่พัฒนาขึ้นจะสามารถทำงานได้เต็มประสิทธิภาพนั้น ทรัพยากรด้านฮาร์ดแวร์มีความสำคัญมาก เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น File server และ Web server ควรจะมีสเปคที่ดีหรือควรเป็นเครื่องแม่ข่าย ไม่ใช่คอมพิวเตอร์ PC ทำงานทั่วไปเพราะ File server ต้องทำหน้าที่บีบอัดและแบ่งไฟล์หรือ Folder พร้อมกับเข้ารหัสลับแบบ AES ทำหน้าที่เป็น Database server และส่งต่อไฟล์ที่ได้ไปยัง Raspberry Pi4 ซึ่งทั้งหมดนี้จะทำงานพร้อม ๆ กัน ส่วนเครื่อง Web server ทำหน้าที่ถอดรหัสลับ AES รวมไฟล์ด้วยโปรแกรม 7zip และให้บริการดาวน์โหลดไฟล์หรือ Folder ผ่านทางเว็บไซต์ ในการให้บริการดาวน์โหลดระบบจำเป็นต้องมีการเตรียมไฟล์ Zip เพื่อดาวน์โหลดไฟล์และ Folder จึงใช้ทรัพยากรมากเพื่อเตรียม การเลือกใช้งานตัวกลางการโอนถ่ายข้อมูลควรเลือกใช้งานคอมพิวเตอร์แทน Raspberry Pi4 เพราะบริการต่าง ๆ ที่รันในตัวของ Raspberry Pi4 มีหลายบริการเช่น เปิดปิด Interface Network, Database server, Network intrusion detection, ส่งไฟล์ข้อมูลไปเครื่อง Web server และ Firewall บริการดังกล่าวข้างต้นจะทำงานพร้อมกันจึงทำให้ Raspberry Pi4 ทำงานหนัก การใช้งานระบบควรใช้สาย Lan ในการเชื่อมต่อ Network ภายในแทน WIFI เพราะสาย Lan มีประสิทธิภาพการโอนถ่ายข้อมูลปริมาณมาก ๆ ไรกว่า WIFI ในขั้นตอนการโอนถ่ายข้อมูลด้วยโพรโทคอล SFTP ควรใช้วิธีเขียนโปรแกรมแบบกระจายการประมวลผลของ CPU เพื่อเพิ่มความเร็วในการโอนถ่ายข้อมูลในงานครั้งนี้ผู้วิจัยยังไม่ได้ปรับแก้ไขตัวอย่างเช่น CPU ของ Raspberry Pi4 ที่ใช้ในการวิจัยครั้งนี้ทำการประมวลผลตัวเดียวจาก CPU ทั้งหมด 4 ตัว ดังภาพ 53 จากภาพ 53 จะเห็นว่าอินเทอร์รัปต์ที่ 31 รับผิดชอบในการส่งข้อมูลและอินเทอร์รัปต์ที่ 32 สำหรับรับข้อมูลซึ่งทำงานบน CPU0 เพียงที่เดียวจึงไม่เหมาะสม ซึ่งผู้วิจัยจะต้องปรับแก้ไขในเวอร์ชันต่อไป การให้บริการดาวน์โหลดผลการวิเคราะห์ข้อมูลยังเป็น Intranet อยู่ซึ่งสามารถใช้งานได้แค่ในมหาวิทยาลัยเท่านั้น เพื่อลดการถูกโจมตี Web server แต่ในอนาคตจะต้องมีแบบ Internet เพื่อให้บริการกับผู้รับบริการจากข้างนอกมหาวิทยาลัยด้วย

```

pi@raspberrypi:~$ cat /proc/interrupts
          CPU0           CPU1           CPU2           CPU3
  9:         0             0             0             0          GICv2 25 Level      vgic
 11: 1330851611 2839097054 2802567112 1916400814  GICv2 30 Level      arch timer
 12:         0             0             0             0          GICv2 27 Level      kvm guest vtimer
 14: 102361796         0             0             0          GICv2 65 Level      fe00b880.mailbox
 15:         344           0             0             0          GICv2 114 Level     DMA IRQ
 16:         363           0             0             0          GICv2 116 Level     DMA IRQ
 17:         0             0             0             0          GICv2 117 Level     DMA IRQ
 23:         0             0             0             0          GICv2 48 Level      arm-pmu
 24:         0             0             0             0          GICv2 49 Level      arm-pmu
 25:         0             0             0             0          GICv2 50 Level      arm-pmu
 26:         0             0             0             0          GICv2 51 Level      arm-pmu
 31: 4221835199         0             0             0          GICv2 189 Level     eth0
 32: 3831493139         0             0             0          GICv2 190 Level     eth0
 34: 1775634821         0             0             0  BRCM STB PCIe MSI 524288 Edge      xhci_hcd
 35:         3707           0             0             0          GICv2 66 Level      VCHIQ doorbell
 36:         7574           0             0             0          GICv2 153 Level     uart-pl011
 37: 73807075         0             0             0          GICv2 158 Level     mmci, mmc0
 38:         0             0             0             0          GICv2 130 Level     febl0000.codec
 39: 56090821         0             0             0          GICv2 106 Level     v3d
 40: 1510263         0             0             0          GICv2 129 Level     vc4 hvs
 41:         229           0             0             0  interrupt-controller@7ef00100 4 Edge      vc4 hdmi hpd connected
 42:         229           0             0             0  interrupt-controller@7ef00100 5 Edge      vc4 hdmi hpd disconnected
 43:         0             0             0             0  interrupt-controller@7ef00100 1 Edge      vc4 hdmi cec rx
 44:         0             0             0             0  interrupt-controller@7ef00100 0 Edge      vc4 hdmi cec tx
 45:         0             0             0             0  interrupt-controller@7ef00100 10 Edge     vc4 hdmi hpd connected
 46:         0             0             0             0  interrupt-controller@7ef00100 11 Edge     vc4 hdmi hpd disconnected
 47:         0             0             0             0  interrupt-controller@7ef00100 7 Edge      vc4 hdmi cec rx
 48:         0             0             0             0  interrupt-controller@7ef00100 8 Edge      vc4 hdmi cec tx
 49:         0             0             0             0          GICv2 107 Level     fe004000.txp
 50:         0             0             0             0          GICv2 141 Level     vc4 crtc
 51:         0             0             0             0          GICv2 142 Level     vc4 crtc, vc4 crtc
 52: 2403351         0             0             0          GICv2 133 Level     vc4 crtc
 53:         0             0             0             0          GICv2 138 Level     vc4 crtc
IPI0: 1988350      5452767      5009669      3738405      Rescheduling interrupts
IPI1:1336879273 2987542582 2917413662 2957518774  Function call interrupts
IPI2:         0             0             0             0          CPU stop interrupts
IPI3:         0             0             0             0          CPU stop (for crash dump) interrupts
IPI4:         0             0             0             0          Timer broadcast interrupts
IPI5: 8925589      21257992     20758228     11603330     IRQ work interrupts
IPI6:         0             0             0             0          CPU wake-up interrupts
Err:         0

```

ภาพ 53 แสดงการกระจายอินเทอร์รัปต์บน CPU ทั้ง 4 ตัวบน Raspberry Pi4



บรรณานุกรม

- 7-zip download. (2021). Retrieved April 21 from <https://www.7-zip.org/download.html>
- Ashibani, Y., & Mahmoud, Q. H. (2017). Cyber physical systems security: Analysis, challenges and solutions. *Computers & Security*, 68, 81-97.
- digitalschool.club. (2565). พื้นฐานการใช้โมดูล *tkinter* สำหรับ *Graphical User Interface (GUI)*. Retrieved 11 พฤศจิกายน from <http://www.digitalschool.club/elearningcom/elearning/python/lesson9/index.php>
- Forcier, J. (2021). *Welcome to Paramiko!* Retrieved April 21 from <http://www.paramiko.org/>
- Gogoi, M. M., Sourav. (2018). DETECTING DDoS ATTACK USING Snort.
- Humayun, M., Jhanjhi, N., Alsayat, A., & Ponnusamy, V. (2021). Internet of things and ransomware: Evolution, mitigation and prevention. *Egyptian Informatics Journal*, 22(1), 105-117.
- kankann. (2563). *Firewall คืออะไร? Firewall มีกี่ประเภท? เปรียบเทียบความแตกต่างของ Firewall*. Retrieved 24 มีนาคม from <https://tips.thaiware.com/1326.html>
- Karahan, O., & Berat, K. (2020). Raspberry Pi firewall and intrusion detection system. *Journal of Intelligent Systems: Theory and Applications*, 3(2), 21-24.
- Khaokaew, N. (2563). สอนเขียน *Python* เพื่อใช้ *SSH* ด้วย *Paramiko* และ *Netmiko* แบบรวบรัด. Retrieved 2 เมษายน from <https://nopnithi.medium.com/สอนการเขียน-python-เพื่อใช้-ssh-ด้วย-paramiko-และ-netmiko-แบบรวบรัด-c3f1a315801b>
- NCSA. (2567). สถิติภัยคุกคามทางไซเบอร์. สืบค้น 8 พฤษภาคม 2567, จาก <https://www.ncsa.or.th/service-statistics.html>
- PoundXI. (2560). *Raspberry Pi คืออะไร ?* Retrieved 22 เมษายน from <https://poundxi.com/raspberry-pi-คืออะไร/>
- Qiu, H., Qiu, M., Liu, M., & Memmi, G. (2020). Secure health data sharing for medical cyber-physical systems for the healthcare 4.0. *IEEE journal of biomedical and health informatics*, 24(9), 2499-2505.
- Raspberrypi.org. (2021). *Raspberry Pi4 model B [image]*. Retrieved April 22 from <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>

SSH.COM. (2021a). *SFTP File Transfer Protocol - get SFTP client & server*. Retrieved April 18 from <https://www.ssh.com/academy/ssh/sftp>

SSH.COM. (2021b). *SSH (Secure Shell) Home Page*. Retrieved April 18 from <https://www.ssh.com/academy/ssh>

SSH.COM. (2021c). *The SSH protocol [image]*. Retrieved April 18 from <https://www.ssh.com/academy/ssh>

ThaiFirewall. (2564). ทำไม่ถึงต้องมี ไฟร์วอลล์. สืบค้น 22 เมษายน 2564, จาก <https://www.thaifirewall.com/whyfirewall.html>

Tongpagdee, R. (2561). *Install Snort on window*. Retrieved 22 เมษายน from <https://medium.com/@rachatatongpagdee/install-snort-on-window-e84b79bcb4b2>

ชัยพร ปานยินดี พุทธภรณ์ เอี่ยมภาณี นิษฐา อรุณสินประเสริฐ. (2560). การรวมกันของวิทยาการอำพรางข้อมูลกับวิทยาการเข้ารหัสลับ สำหรับภาพทางการแพทย์. วารสารวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ, 12(1), 20-32.

นพดล จินตสุนทรอุไร ตีรรัตน์ เมตต์การุณจิต. (2558). ราชภัฏเบอร์ฟาย ไฟร์วอลล์ สำหรับหน่วยงานธุรกิจขนาดกลางและขนาดเล็ก (SMEs). *Journal of Business Administration and Languages (JBAL)*, 3(2), 20-25.

มีทรัพย์หลาก, ก. (2562). "ระบบไซเบอร์-กายภาพ" พื้นฐานสำคัญในการยกระดับเทคโนโลยี. Retrieved 1 กุมภาพันธ์ from <https://www.nectec.or.th/research/research-project/nectec-cps.html>