รายงานสหกิจศึกษาฉบับสมบูรณ์

ตรวจจับข้อมูลบนเครือข่ายด้วย WiFi Pineapple

Matching pattern on network traffic with WiFi Pineapple

นายเมธาสิทธิ์ รินทร

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

รายงานสหกิจศึกษาฉบับสมบูรณ์

ตรวจจับข้อมูลบนเครือข่ายด้วย WiFi Pineapple

Matching pattern on network traffic with WiFi Pineapple

นายเมธาสิทธิ์ รินทร

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

ชื่อโครงงานสหกิจศึกษา (ภาษาไทย) ตรวจจับข้อมูลบนเครือข่ายด้วย WiFi Pineapple

ชื่อ-สกุล นักศึกษา นายเมธาสิทธิ์ รินทร

คณะ วิศวกรรมศาสตร์ **ภาควิชา** วิศวกรรมคอมพิวเตอร์

ชื่อ-สกุล อาจารย์นิเทศ นายจิระศักดิ์ สิทธิการ และ นายบัณฑิต พัสยา

ชื่อ-สกุล ผู้นิเทศงาน นายพิชญะ โมริโมโต

สถานประกอบการ บริษัท เซค คอนซัลท์ (ไทยแลนด์) จำกัด

บทคัดย่อ

ภัยคุกคามจากคอมพิวเตอร์ในปัจจุบัน เกิดขึ้นอย่างรวดเร็วและหลากหลาย ไม่ว่าจะเป็นการส่งข้อมูลมหาศาลเพื่อทำให้ระบบใช้งานไม่ได้ การโจมตีผ่านเว็บไซต์เพื่อยึดครองระบบ การแฝงตัวเข้าไปในองค์กรเพื่อขโมยข้อมูล แต่อีกหนึ่งภัยคุกคามต่อองค์กรที่เกิดขึ้นได้ง่าย นั่นคือ การที่พนักงานในองค์กรเชื่อมต่อเครือข่ายไร้สายที่เกิดจากผู้ไม่หวังดีสร้างขึ้นมา เพื่อหวังขโมยข้อมูลจากผู้ใช้งาน การทดสอบเจาะระบบเครือข่ายไร้สาย จึงช่วยให้พนักงานในองค์กรได้ตระหนักถึงภัยคุกคามจากการใช้งานเครือข่ายที่ไม่ปลอดภัย ทั้งที่มามาจากภายนอกและภายในขององค์กรเอง

WiFi Pineapple คือชุดอุปกรณ์ที่รวบรวมเครื่องมือ เพื่อใช้ทดสอบระบบความปลอดภัยของเครือข่ายไร้สาย ความสามารถหลักคือ ตัวอุปกรณ์สามารถสร้าง Access point ขึ้นมา และดักจับข้อมูลบนเครือข่ายของผู้ใช้ที่ได้เชื่อมต่อกับ WiFi Pineapple เอาไว้ โดยความสามารถเพิ่มเติมอื่น ๆ นั้น จะขึ้นอยู่กับโมดูลที่ได้ติดตั้งเอาไว้ ซึ่งสามารถดาวน์โหลดได้จากเว็บไซต์หลักเอง หรือสามารถเขียนขึ้นเองก็ได้เช่นกัน

เนื่องจากข้อมูลที่วิ่งอยู่บนเครือข่ายนั้นมีปริมาณมาก ทำให้ยากต่อการค้นหาสิ่งที่น่าจะเป็นประโยชน์ต่อการทดสอบเจาะระบบ จึงได้มีการสร้างโมดูลที่ช่วยค้นหาข้อมูลให้ตรงกับความต้องการมากขึ้น และนำผลลัพธ์จากการค้นหานั้นแสดงผลบนหน้าเว็บไซต์

คำสำคัญ : ความปลอดภัยระบบเครือข่าย, WiFi Pineapple, ngrep

Cooperative Title: Matching pattern on network traffic with WiFi Pineapple

Student intern name: Metasit Rinthon

Faculty: Engineering **Department:** Computer Engineering

Advisor name: Jirasak Sittigorn and Bundit Pasaya

Mentor name: Pichaya Morimoto

Company: SEC Consult (Thailand) Co.,Ltd.

ABSTRACT

Nowadays computer threats are varied and happened quickly. Such as Denied-of-Service attack, compromise system by website hacking, advance persistent threat and one type of attack that causes harm to organize is a rogue access point. An inside information could be stolen if employees not aware of this attack. Wireless penetration testing could help employees aware and avoid using the wireless networks which not secure.

WiFi Pineapple is a collection of tools for wireless audit platform, the main feature is creating rogue AP and capture packets on the network, additional modules can be installed by download from a repository or write own module.

There are many packets on network traffic which difficult to find interesting information. So, the module can search patterns to meet more needs, the results will show in a webpage on the device.

Keywords : Security, WiFi Pineapple, ngrep

กิตติกรรมประกาศ

รายงานสหกิจศึกษาฉบับสมบูรณ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดีด้วยความช่วยเหลือจากหลาย ๆ ฝ่าย ทั้งในทางตรงและทางอ้อม ซึ่งรายงานฉบับนี้จะสำเร็จลุล่วงไม่ได้เลยหากปราศจากความช่วยเหลือของบุคคลเหล่านี้

บริษัท เซค คอนซัลท์ (ไทยแลนด์) จำกัด ที่เปิดโอกาสรับนักศึกษาเข้ามาฝึกงาน พี่เก่ง ที่คอยช่วยจัดการงานในด้านต่าง ๆ พี่ตะ พี่หลง และพี่ท็อป ที่คอยช่วยสอนงาน และให้คำแนะนำทั้งในด้านการทำงานและเรื่องทั่วไป ทำให้ปรับตัวเข้ากับทีมได้เร็วขึ้น

ขอขอบพระคุณอาจารย์และบุคลากรทุกท่านในภาควิชาวิศวกรรมคอมพิวเตอร์ ที่ได้ช่วยสอนสั่งความรู้ต่าง ๆ มาโดยตลอด และให้โอกาสนักศึกษาเข้าร่วมโครงการสหกิจศึกษา

ขอบคุณเพื่อน ๆ พี่ ๆ น้อง ๆ รวมถึงผู้ที่เกี่ยวข้องที่ไม่ได้กล่าวถึงไว้ ณ ที่นี้ ที่คอยให้คำปรึกษา ช่วยแก้ปัญหาในด้านต่าง ๆ

ท้ายที่สุดนี้ขอกราบขอบพระคุณบิดา มารดา และครอบครัวที่ได้เลี้ยงดูสั่งสอน พร้อมทั้งให้โอกาสในการศึกษาและให้กำลังใจเสมอมา คุณประโยชน์ใดที่มีจากรายงานฉบับนี้ ขอมอบให้แด่ผู้มีพระคุณทุกท่านที่ได้กล่าวมา ขอขอบพระคุณมา ณ โอกาสนี้

เมธาสิทธิ์ รินทร

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญภาพ

ภาพที่ **หน้า**

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

ในกระบวนการทดสอบการเจาะระบบนั้น แบ่งออกเป็น 5 ขั้นตอน ดังนี้

1) Reconnaissance คือกระบวนการเก็บรวบรวมข้อมูลของเป้าหมาย เพื่อใช้ในการวางแผนการเจาะระบบในขั้นตอนต่อไป ในกระบวนการเก็บข้อมูล สามารถแบ่งย่อยได้เป็น 2 แบบ คือ 1.1) Actively คือส่งคำสั่งบางอย่าง เพื่อกระทำต่อเครื่องเป้าหมายโดยตรง 2.2) Passively คือการหาข้อมูลของเป้าหมายทางอ้อม เช่นค้นหาจาก Search engine

2) Scanning ในกระบวนการนี้ จะต้องอาศัยเครื่องมือที่ช่วยในการแสกนเป้าหมาย เช่น ใช้เครื่องมือจำพวก vulnerability scanner เพื่อค้นหาช่องโหว่ที่เปิดเผยสู่สาธารณะแล้ว

3) Gaining Access คือกระบวนการเข้าควบคุมส่วนใดส่วนหนึ่งของเป้าหมายหรือเครือข่ายเป้าหมาย เช่น การเจาะระบบผ่านเว็บไซต์ต่างโดเมนแต่อยู่ในเครื่องเดียวกัน เพื่อดึงข้อมูลออกมา

4) Maintaining Access คือการรักษาช่องทางที่ใช้ในการเข้าควบคุมเครื่องเป้าหมาย หลังจากที่นักทดสอบระบบเข้าควบคุมเครื่องเป้าหมายได้แล้ว ในครั้งต่อไปที่จะเข้าถึงข้อมูลส่วนนี้ นักทดสอบระบบไม่จำเป็นที่จะต้องหลบหลีกการตรวจจับการโจมตีต่าง ๆ อีก แต่จะใช้วิธีวาง backdoor เอาไว้ เช่น หลังจากเจาะระบบผ่านเว็บไซต์ได้ นักทดสอบระบบจะอัพโหลดไฟล์ PHP shell และในครั้งต่อไปที่จะเข้าควบคุมเครื่อง ก็จะเข้าผ่าน PHP shell ที่วางไว้

5) Covering Tracks ในขั้นตอนสุดท้ายของกระบวนการทดสอบเจาะระบบ คือการลบร่องรอยต่าง ๆ ที่นักทดสอบระบบได้ทำไว้ เช่น ประวัติการ Login, Access Log, Error log ทุก ๆ ประวัติการรันคำสั่ง จะต้องถูกลบทิ้ง เพื่อไม่ให้แอดมินของเครื่องตรวจจับได้

สำหรับกรณีที่การทดสอบเจาะระบบ ได้ระบุให้กระทำในสถานที่ขององค์กรจริง ๆ ในขั้นตอนแรก (Reconnaissance) นั้น นอกเหนือจากการรวบรวมข้อมูลแบบเทคนิคคอลแล้ว นักทดสอบระบบยังสามารถใช้วิธี Social engineering เพื่อเก็บข้อมูลจากพนักงานขององค์กร และวิธีการที่จะใช้เก็บข้อมูลนั้น การทำ Rogue Access Point ก็เป็นวิธีที่น่าสนใจ เนื่องจากตรวจสอบได้ยากว่า Access Point ที่ใช้กระจายสัญญาณอินเตอร์เน็ตนั้น เป็นขององค์จริง ๆ หรือไม่ และวิธีการนี้ ยังช่วยให้องค์กรเพิ่มความตระหนักรู้ให้กับพนักงาน ให้ได้รู้ถึงภัยคุมคามจากการเชื่อมต่อ Access Point ที่ไม่น่าปลอดภัย

WiFi Pineapple [1] คืออุปกรณ์ที่ช่วยทดสอบระบบความปลอดภัยของเครือข่ายไร้สายด้วยเทคนิคต่าง ๆ จากบริษัท Hak5 โดยมีตัว Web interface มาช่วยให้นักทดสอบระบบใช้งานได้ง่ายขึ้น ในปัจจุบันได้ออกรุ่นที่ 6 โดยฮาร์ดแวร์มีสองเวอร์ชั่น คือ 1) NANO จะมีขนาดเล็ก มีเสาสัญญาณ 2 เสา และ 2) TETRA มีขนาดใหญ่กว่า มีเสาสัญญาณ 4 เสา

ภาพที่ 1.1 อุปกรณ์ WiFi Pineapple ทั้งสองเวอร์ชั่น

คุณสมบัติหลักของ WiFi Pineapple มีชื่อว่า PineAP ใช้สร้าง Access Point ของปลอมขึ้นมาเพื่อหลอกให้ผู้ใช้เชื่อมต่อ อีกทั้งยังจำกัดอุปกรณ์ของเป้าหมายได้ด้วยการทำ Blacklist และ Whitelist ส่วนความสามารถอื่น ๆ จะขึ้นอยู่กับ Module ที่ติดตั้งเอาไว้

ข้อมูลที่วิ่งอยู่บนโปรโตคอล HTTP มีจำนวนมาก โครงงานนี้จึงนำเสนอแนวทางการกรองข้อมูลที่น่าจะมีความสำคัญต่อกระบวนการทดสอบสอบเจาะระบบ เช่น credentials ของพนักงานในองค์กรที่ใช้บนอินเตอร์เน็ตอาจจะตรงกับที่ใช้ภายในองค์กร โดยใช้คำสั่ง ngrep ค้นหาตาม pattern ที่ต้องการ และแสดงผลลัพธ์ผ่านทาง Web interface

1.2 วัตถุประสงค์ของการวิจัย

1) เพื่อให้สามารถสังเกตข้อมูลที่สนใจได้ง่ายขึ้น
2) เพื่อศึกษาการพัฒนาโมดูลเพื่อใช้บน WiFi Pineapple
3) เพื่อเพิ่มความสะดวกให้นักทดสอบระบบ ไม่ต้องรันคำสั่งผ่าน command line

1.3 ขอบเขตของการวิจัย

1) พัฒนาโมดูลโดยใช้ framework ที่มีอยู่ใน WiFi Pineapple อยู่แล้ว
2) ข้อมูลนำเข้าที่ใช้วิเคราะห์ คือข้อมูลที่วิ่งอยู่บนโปรโตคอล HTTP
3) โมดูลที่พัฒนา สามารถทำงานได้บน WiFi Pineapple TETRA
4) บันทึกผลที่ได้จากการวิเคราะห์ลงไนไฟล์ และเก็บไว้ใน Internal storage

1.4 วิธีการดำเนินการวิจัย

1) เข้าใจคำสั่ง command line เบื้องหลังของ module ที่จะพัฒนา

2) ออกแบบรูปแบบการนำผลลัพธ์มาแสดงผ่านหน้าเว็บไซต์

3) ศึกษา workflow การเรียกใช้โมดูลของ WiFi Pineapple

4) ศึกษาคำสั่งต่าง ๆ ที่ใช้ จากโมดูลหลักที่มีให้ดาวน์โหลด

5) ดำเนินการสร้างและทดสอบโมดูล

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1) เข้าใจกระบวนการพัฒนาโมดูลของ WiFi Pineapple

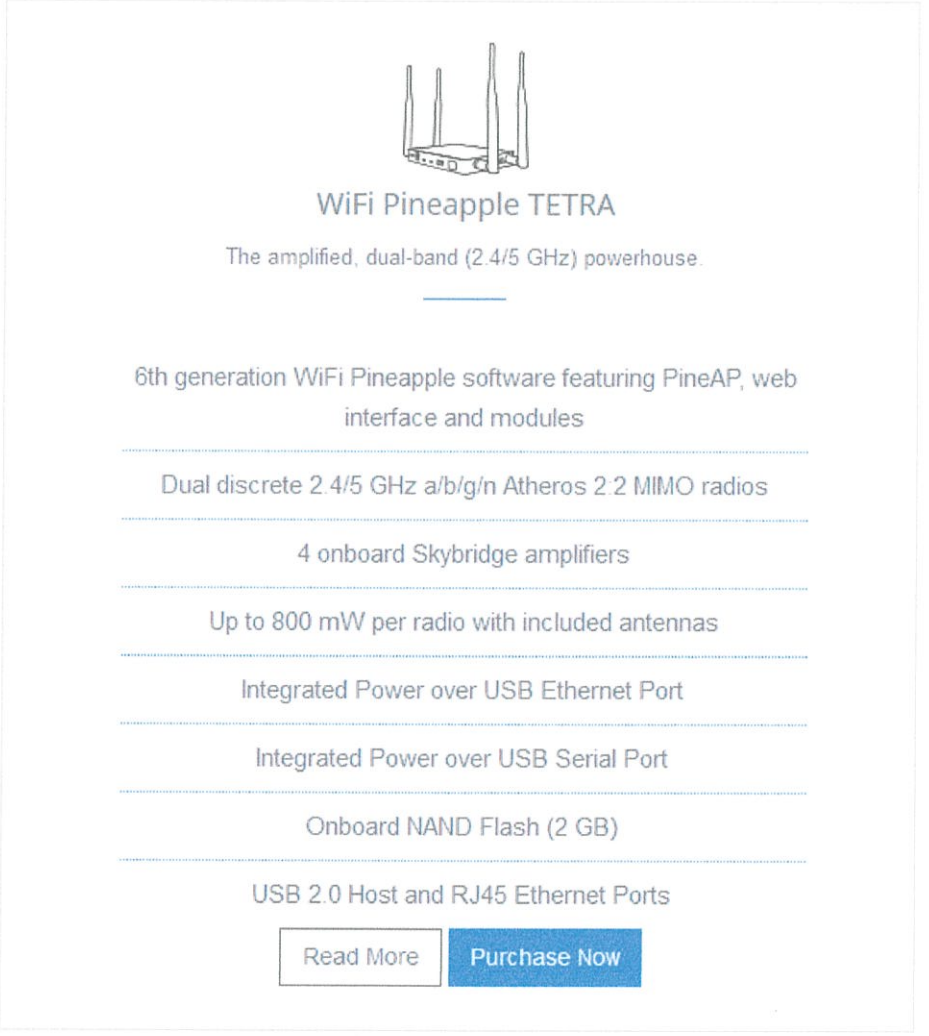
2) สามารถแก้ไขโมดูลอื่น เพื่อให้การใช้งานตรงตามความต้องการมากขึ้น

3) สามารถใช้งานได้จริง และช่วยลดเวลาในการวิเคราะห์ข้อมูลที่วิ่งบนเครือข่าย

บทที่ 2
แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 WiFi Pineapple TETRA

เป็นอุปกรณ์หลักที่ใช้ในโครงงานนี้ โดยมีคุณสมบัติด้านฮาร์ดแวร์ดังรูปด้านล่างนี้



ภาพที่ 2.1 รายละเอียดฮาร์ดแวร์

WiFi Pineapple ทำงานอยู่บนระบบปฏิบัติการ OpenWrt โดยสามารถเรียกใช้ผ่าน Web interface มีคุณสมบัติคร่าว ๆ ดังนี้

- Dashboard แสดงภาพรวมว่ามีอุปกรณ์ใดเชื่อมต่อ WiFi Pineapple อยู่ และ WiFi Pineapple เอง กระจายสัญญาณ Wireless ในชื่อใดบ้าง

- Recon ค้นหา Access Point อื่น ๆ และ Client devices ที่อยู่รอบ ๆ

- Module สำหรับดาวน์โหลดและติดตั้ง module เสริมจาก Official website

- PineAP เป็นซอฟต์แวร์หลัก มีหน้าที่กระจายสัญญาณ Access Point และดักจับ Access Point อื่น ๆ

- Tracking เพื่อติดตาม Client devices แบบเฉพาะเจาะจง ว่าเชื่อมต่อกับ Access Point ใดบ้าง และส่ง Probe request ใดออกมาบ้าง

ในขั้นตอนของการติดตั้ง สามารถศึกษาเพิ่มเติมได้จาก WiFi Pineapple Wiki [2]

2.2 OpenWrt

OpenWrt [3] คือ ระบบปฏิบัติการแบบ Open source มีพื้นฐานมาจาก Linux สำหรับติดตั้งบนอุปกรณ์ Embedded จำพวก Router เพื่อใช้จัดการ Network traffic โดยตั้งค่าผ่าน Command-Line interface หรือ Web interface โดยซอฟต์แวร์ต่าง ๆ จะถูกติดตั้งโดยผ่าน Package manager ที่ชื่อว่า opkg

2.3 ngrep command

ngrep ย่อมาจาก network-grep คือคำสั่งบน linux ที่ใช้ค้นหาข้อความตาม pattern ที่กำหนด มีลักษณะการทำงานคล้ายคลึงกับคำสั่ง grep ที่ใช้ค้นหาข้อความจาก STDIN หรือ STDOUT เพียงแต่คำสั่ง ngrep จะค้นหาจาก Network interface หรือจาก PCAP file

พารามิเตอร์ที่น่าสนใจ ได้แก่

-q (quite) เพื่อซ่อนผลลัพธ์ที่ไม่ต้องการ

-i (case-insensitive) ค้นหา pattern ที่กำหนด โดยไม่สนใจตัวพิมพ์เล็กหรือพิมพ์ใหญ่

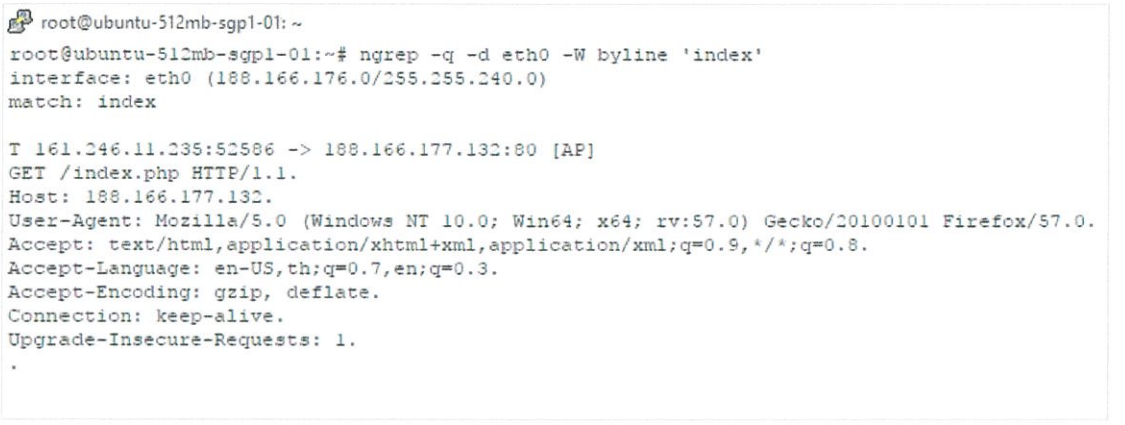
-d (device) กำหนด Network interface ต้นทางที่จะใช้ค้นหา

-W กำหนดรูปแบบของผลลัพธ์ โดยมีค่าให้เลือกใช้ ได้แก่ normal, byline, single, none

ตัวอย่างคำสั่งที่ใช้ค้นหา TCP connections ที่มีคำว่า “index” จาก Network interface eth0 โดยซ่อน packets อื่นที่ไม่เกี่ยวข้อง

ตัวอย่าง 2.1 การใช้งานคำสั่ง ngrep

```
User@Host:~# ngrep -q -d eth0 -W byline 'index'
```



ภาพที่ 2.2 ตัวอย่างการใช้งานคำสั่ง ngrep

2.4 Languages and Frameworks

ภาษาหลักที่ใช้ในการพัฒนาโมดูลบน WiFi Pineapple ได้แก่ PHP, HTML, Javascript โดยใช้ AngularJS framework ช่วยพัฒนา

AngularJS คือ framework สำหรับพัฒนาแอพลิเคชั่นในฝั่ง Client ในรูปแบบของ Javascript

PHP คือ ภาษาที่ใช้ประมวลผลคำสั่งอยู่เบื้องหลัง

ตัวอย่าง 2.2 เวอร์ชั่นของ PHP และ AngularJS

```
root@Pineapple:/# php-fpm -v
PHP 5.6.17 (fpm-fcgi) (built: Oct 28 2016 05:45:39)
Copyright (c) 1997-2015 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2015 Zend Technologies
root@Pineapple:/# head -n 5 /pineapple/js/vendor/angular.min.js
/*
 AngularJS v1.4.5
 (c) 2010-2015 Google, Inc. http://angularjs.org
 License: MIT
*/
```

บทที่ 3
วิธีดำเนินการวิจัย

3.1 ความต้องการของระบบ

1) โมดูลที่จะพัฒนาขึ้นมา ต้องสามารถติดตั้งได้บน WiFi Pineapple TETRA

2) มีเครื่องมือ ngrep ติดตั้งไว้บนระบบปฏิบัติการ ซึ่งระบบปฏิบัติการ OpenWrt ติดตั้งคำสั่ง ngrep มาโดยพื้นฐานอยู่แล้ว สามารถตรวจได้โดยรันคำสั่ง

ตัวอย่าง 3.1 แสดงเวอร์ชั่นของ ngrep

```
root@Pineapple:~# ngrep -V
ngrep: V1.45, $Revision: 1.93 $
```

3.2 การออกแบบ

ขั้นตอนการเรียกใช้ Module ของ WiFi Pineapple มีลักษณะดังนี้

1) เมื่อเปิดโมดูลหรือคลิกปุ่มต่าง ๆ AngularJS จะส่ง request ไปยัง PHP

2) PHP ประมวลผลคำสั่ง เช่น แสดงข้อความ, รันคำสั่งไว้เบื้องหลัง ถ้ามีผลลัพธ์ ผลลัพธ์จะถูกส่งกลับไปยัง AngularJS อีกรอบ

3) HTML จะเรียกใช้ AngularJS และนำข้อความที่ได้มาแสดงผลผ่านหน้าเว็บไซต์

ทุกโมดูลใด ๆ จะประกอบไปด้วย 4 ไฟล์ที่จำเป็น ได้แก่ 1) module.html ประกอบไปด้วยภาษา HTML ที่ใช้แสดงผลบนหน้าเว็บ 2) module.info ประกอบไปด้วยคำรายละเอียดของโมดูล เช่น ชื่อโมดูล เวอร์ชั่น ชื่อผู้พัฒนา คำอธิบาย ในรูปแบบ JSON 3) module.js จะอยู่ในโฟลเดอร์ js/ ซึ่งใช้เก็บโค้ดของ AngularJS 4) module.php จะอยู่ในโฟลเดอร์ api/ เพื่อเก็บโค้ดภาษา PHP ที่ใช้ในการประมวลผล โครงสร้างของไฟล์ที่กล่าวมามีลักษณะดังนี้

ตัวอย่าง 3.2 โครงสร้างไฟล์ของโมดูล

```
.
├── js
│   └── module.js
├── module.html
├── module.info
└── api
    └── module.php
```

โดยทุก ๆ โมดูลจะถูกเก็บไว้ใน /pineapple/modules/

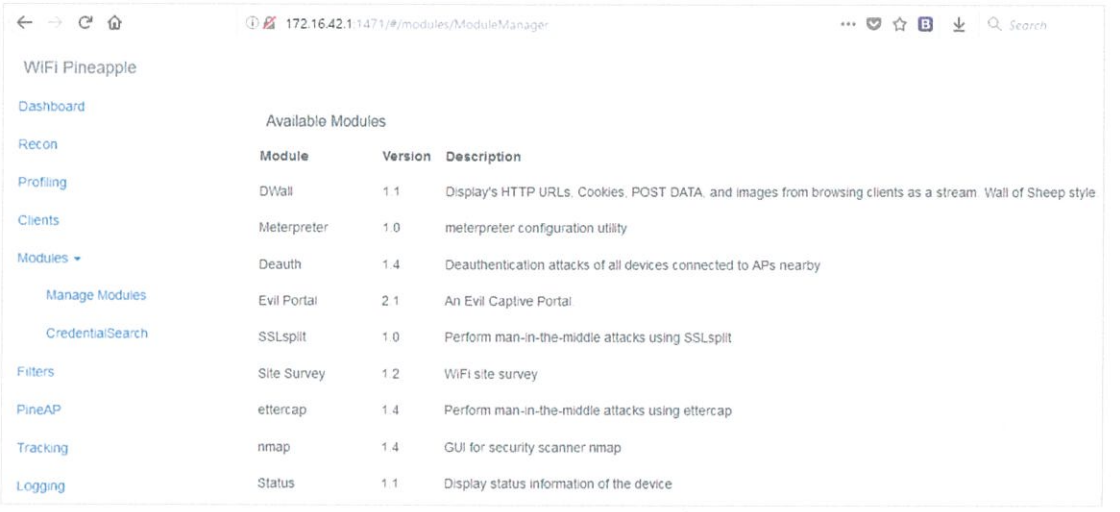
ความสามารถของโมดูลที่จะพัฒนา ได้แก่ มี pattern ที่ใช้กรองข้อมูลที่หลากหลายและสามารถบันทึกประวัติการกรองเอาไว้ เพื่อมาย้อนเปิดดูในภายหลังได้ ดังนั้นจึงได้ออกแบบให้ไฟล์ที่เก็บชุดของ pattern ที่ใช้ค้นหาข้อความ จะอยู่ในโฟลเดอร์ profiles/ และประวัติของการผลลัพธ์ทั้งหมด จะอยู่ในโฟลเดอร์ log/

3.3 การดำเนินการ

WiFi Pineapple เอง มีโมดูลที่ช่วยสร้างโมดูลในเบื้องต้นให้พร้อมกับโค้ดตัวอย่าง ชื่อว่า Module Maker ซึ่งสามารถดาวน์โหลดและติดตั้งได้ผ่าน Web interface โดยการไปที่ http://172.16.42.1:1471 ซึ่งเป็น Default IP ของ WiFi Pineapple เอง และคลิกไปที่เมนู

Modules > Manage Modules > Get Modules from WiFiPineapple.com

และเลือก Install ที่ Module Maker หลังจากที่ดาวน์โหลดและติดตั้งเรียบร้อยแล้ว จะได้ไฟล์ตามตัวอย่างที่ 3.2 และเนื้อหาของไฟล์ module.info จะถูกกำหนดมาให้ จากการตั้งค่าใน Module Maker



ภาพที่ 3.1 Web interface ที่ใช้ติดตั้งโมดูล

```
{
    "title": "CredentialSearch",
    "description": "For searching credentials in HTTP base on ngrep",
    "version": "1.0",
    "author": "Metasit Rinthon"
}
```

ภาพที่ 3.2 เนื้อหาของไฟล์ module.info

เพื่อความสะดวกในการพัฒนาโมดูล ดังนั้นโค้ดต่าง ๆ จึงควรดาวน์โหลดมาเก็บไว้ในเครื่อง Local เสียก่อน

ในส่วนของไฟล์ module.html จะใช้ Bootstrap ในการตกแต่ง และเนื่องจากโมดูลถูกเขียนโดย AngularJS ดังนั้นไฟล์ HTML จึงไปดึงค่ามาจาก AngularJS ตามตัวอย่างรูป 3.3 คือโค้ดที่ใช้ดึงค่า title และ version จาก AngularJS มาแสดงผลบนหน้าเว็บ โดยการเรียกใช้ tag ด้วย argument ng-controller="ControllerName" โดยในโค้ดนี้ ControllerName คือ CredentialSearch_Controller

```
<div class="panel panel-default" ng-controller="CredentialSearch_Controller">
    <div class="panel-heading">
        <h4 class="panel-title pull-left">{{title}}</h4>
        <span class="pull-right">{{version}}</span>
        <div class="clearfix"></div>
    </div>
</div>
```

ภาพที่ 3.3 เนื้อหา module.html บางส่วน

ไฟล์ที่เก็บโค้ดของ AngularJS คือ js/module.js เริ่มจากการเรียกใช้ฟังก์ชัน built-in API registerController(); ตามรูปตัวอย่าง 3.4 ชื่อของ Controller คือ CredentialSearch_Controller และ dependencies ที่มักจะเรียกใช้ ได้แก่ $api และ $scope

เพิ่มตัวแปรภายใน scope ที่มีชื่อว่า title และ version ด้วยการประกาศตัวแปรตามตัวอย่างด้านล่าง

ตัวอย่าง 3.3 ประกาศตัวแปร title และ version ใน AngularJS

```
$scope.title = response.title;
$scope.version = "Version "+response.version;
```

การส่งข้อมูลไปยัง PHP ทำได้โดยเรียกใช้ฟังก์ชัน $api.request() โดยจะส่ง request ไปยัง module.php และรับ response มากำหนดค่า $scope.title ให้เป็น response.title และกำหนดค่า $scope.version ให้เป็น response.version

```
registerController('CredentialSearch_Controller', ['$api', '$scope', function($api, $scope) {
    /* It is good practice to 'initialize' your variables with nothing */
    $scope.greeting = "";
    $scope.content = "";
    $scope.title = "Loading...";
    $scope.version = "Loading...";

    /* Use the API to send a request to your module.php */
    $scope.getHeader = (function() {
    $api.request({
        module: 'CredentialSearch', //Your module name
        action: 'getHeader'   //Your action defined in module.php
    }, function(response) {
        $scope.title = response.title;
        $scope.version = "Version "+response.version;
      })
    });

    $scope.getHeader();

}]);
```

ภาพที่ 3.4 เนื้อหา module.js ที่ใช้แสดง title และ version

ในส่วนของไฟล์ api/module.php จะประกอบไปด้วยโค้ด PHP ทั้งหมดที่โมดูลเรียกใช้งาน เริ่มต้นด้วยการสร้าง class ที่ extends มาจาก Module ซึ่งต้องระบุไว้ภายใน namespace ที่ชื่อ pineapple และสร้าง method ขึ้นมาเพื่อใช้ในการจัดการ request

```php
<?php namespace pineapple;
class CredentialSearch extends Module
{
    public function route()
    {
        switch ($this->request->action) {
            case 'getContents':    // If you request the action "getContents" from your Javascript,
            this is where the PHP will see it, and use the correct function
                $this->getContents();  // $this->getContents(); refers to your private function
                that contains all of the code for your request.
                break;                 // Break here, and add more cases after that for different
                requests.
            case 'getHeader':
                $this->getHeader();
                break;
        }
    }
```

ภาพที่ 3.5 method route() ที่ใช้จัดการ request

โค้ดตามตัวอย่างในรูปที่ 3.5 นี้ จะทำงานทุก ๆ action ที่รับมาจาก Javascript ซึ่งจากรูปที่ 3.4 คือ getHeader เมื่อ PHP เจอ getHeader จะทำการวิ่งไปประมวลผลในฟังก์ชัน getHeader()

```php
private function getHeader()
{
  $moduleInfo = @json_decode(file_get_contents("/pineapple/modules/CredentialSearch/module.info"));
  $this->response = array('title' => $moduleInfo->title, 'version' => $moduleInfo->version);
}
```

ภาพที่ 3.6 ฟังก์ชัน getHeader()

เมื่อฟังก์ชัน getHeader ถูกประมวลผล จะไปอ่านไฟล์ module.info ซึ่งเป็น JSON แล้วนำมาแปลงกลับ หลังจากนั้นนำ title และ module ส่งค่า response กลับไปให้ AngularJS

โค้ดในส่วนของการแสดง title และ version ทั้งหมด จึงเป็นไปตามรูปที่ 3.7

```php
<?php namespace pineapple;
class CredentialSearch extends Module
{
    public function route()
    {
        switch ($this->request->action) {
            case 'getHeader':
                $this->getHeader();
                break;
        }
    }

    private function getHeader()
    {
      $moduleInfo = @json_decode(file_get_contents("/pineapple/modules/CredentialSearch/module.info"));
      $this->response = array('title' => $moduleInfo->title, 'version' => $moduleInfo->version);
    }
}
```

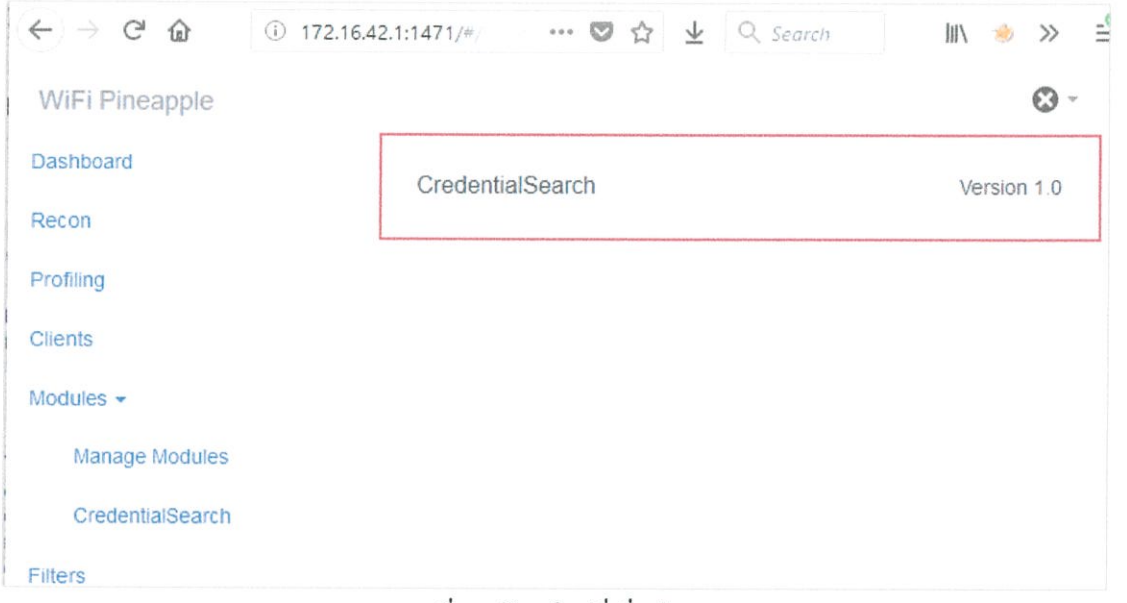
ภาพที่ 3.7 PHP code ที่ใช้แสดง title และ version

เมื่อเขียนไฟล์ทั้งหมดที่จะใช้แสดง version และ title ครบแล้ว (HTML, JS, PHP) ลำดับต่อมาคือการอัพโหลดไฟล์ทั้งหมด ขึ้นไปยัง WiFi Pineapple ผ่านทางคำสั่ง scp [4] บน linux

ตัวอย่าง 3.4 คำสั่ง scp เพื่ออัพโหลดไฟล์

```
root@Pineapple:~# scp -r CredentialSearch/ root@172.16.42.1:/pineapple/modules/.
```

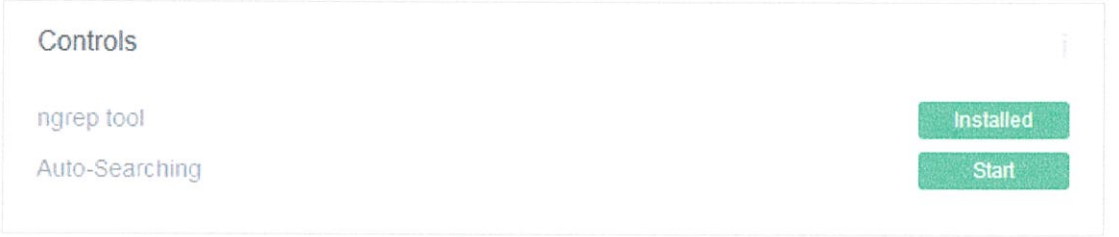
ขั้นต่อมา คือการตรวจสอบดูว่าโมดูลที่เราเพิ่งอัพโหลดไปนั้น แสดงผลได้ถูกต้องหรือไม่ โดยการเปิดไปที่ http://172.16.42.1:1471 ซึ่งเป็นหน้า management ของ WiFi Pineapple หลังจากที่ล็อกอินเรียบร้อย ให้สังเกตที่เมนู Modules ว่ามีโมดูลตัวใหม่ที่เราเพิ่งสร้างไปปรากฏอยู่หรือไม่



ภาพที่ 3.8 โมดูลใหม่ที่เพิ่มเข้ามา

จะพบว่าโมดูล CredentialSearch ปรากฏขึ้นมา พร้อมกับแสดง title และ version ได้อย่างถูกต้อง

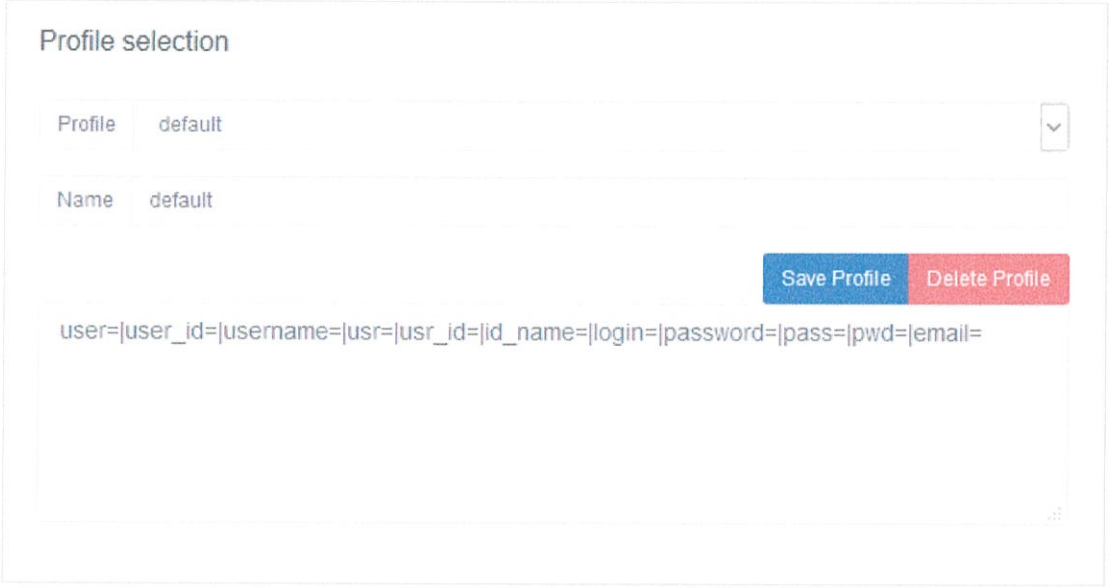
หลังจากที่สร้างส่วนหัวของโมดูลแล้ว ส่วนต่อมา คือ ส่วนที่เรียกว่า Control มีหน้าที่เช็คว่า WiFi Pineapple มีคำสั่ง ngrep ติดตั้งเอาไว้หรือไม่ ถ้าติดตั้งไว้แล้ว จะสามารถเริ่มการทำงานได้จากส่วนนี้



ภาพที่ 3.9 หน้าต่าง Controls

หน้าต่างถัดมา คือ Profile selection มีหน้าที่แก้ไขโปรไฟล์ บันทึกโปรไฟล์ และลบโปรไฟล์ โดยโปรไฟล์เสมือนกับว่าเป็น Pattern ที่เราจะใช้ค้นหา ทุกครั้งที่มีการบันทึก หรือเลือกโปรไฟล์ใหม่ จะเป็นการเปลี่ยนแปลงค่า $rootScope.profileData เพื่อให้ registerController อื่นเรียกใช้งาน

$rootScope.profileData หมายถึง ข้อมูลของ Profile นั้น ๆ โดยสามารถบันทึกไว้ในรูปแบบ Regular Expression ได้



ภาพที่ 3.10 หน้าต่าง Profile selection

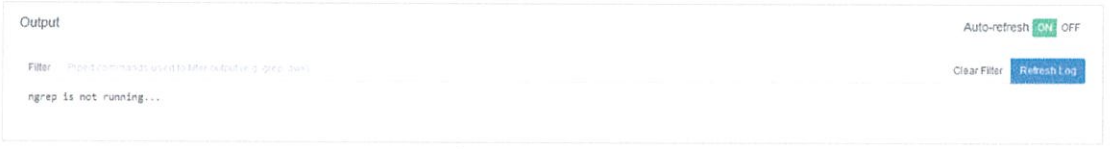
เนื่องจากต้องการรับข้อความที่ต้องการค้นหาผ่านทางเว็บไซต์เพียงอย่างเดียว ดังนั้น Network interface และพารามิเตอร์อื่น ๆ จะถูกกำหนดไว้ไม่ให้แก้ไข ในส่วนของคำสั่งที่อยู่เบื้องหลังการทำงานนั้นจะใช้คำสั่ง ngrep และค่าที่ใช้ค้นหานำมาจาก $rootScop.profileData รูปแบบของคำสั่งเป็นไปตามตัวอย่างต่อไปนี้

ตัวอย่าง 3.5 คำสั่ง ngrep ที่ใช้งาน

```
root@Pineapple:~# ngrep -i -d wlan0 -W byline "ProfileData" -O output.pcap >> output.log
```

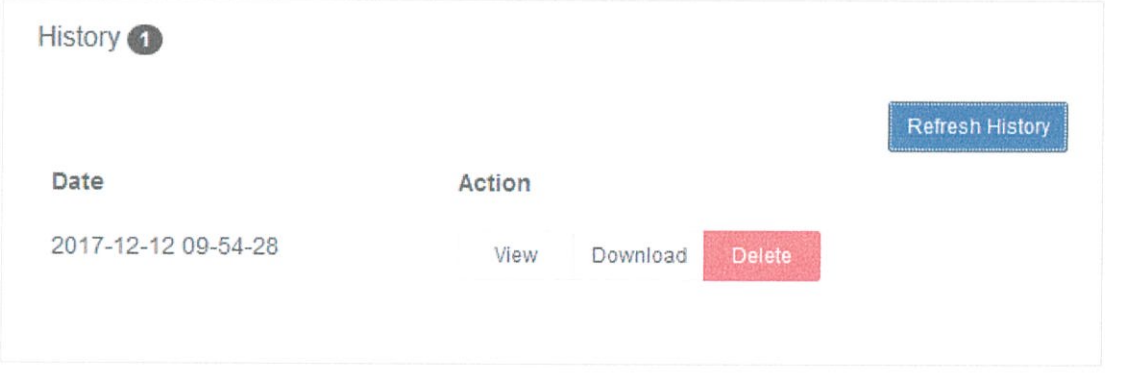
ผลลัพธ์จากการรันคำสั่ง ngrep จะถูกบันทึกลงไปในไฟล์ที่อยู่ในโฟลเดอร์ log/

หน้าต่างถัดมา คือ Output มีหน้าที่แสดงผลลัพธ์โดยการอ่านไฟล์ โดยยึดตามเวลาที่สร้างไฟล์ล่าสุดที่อยู่ในโฟลเดอร์ log/ โดยมีฟังก์ชัน Auto-refresh กำหนดเวลาเมื่อครบทุก ๆ 5 วินาที ไฟล์ PHP จะทำการอ่านไฟล์ใน log/ ทุกครั้ง



ภาพที่ 3.11 หน้าต่าง Output

สุดท้ายคือหน้าต่าง History มีไว้แสดงไฟล์ที่บันทึกไว้ใน log/ โดยสามารถคลิกเพื่อดาวน์โหลดไฟล์ ลบไฟล์ และอ่านไฟล์บน Browser ได้เลย



ภาพที่ 3.12 หน้าต่าง History

ฟังก์ชันทั้งหมด ที่ถูกประกาศไว้ใน method route() ได้แก่

1) getHeader() ใช้แสดง title และ version

2) getProfiles() ใช้แสดงรายชื่อ profile ทั้งหมด

3) showProfile() ใช้แสดงเนื้อ profile ที่ถูกเลือก

4) deleteProfile() ใช้ลบ profile

5) saveProfileData() ใช้บันทึก profile

6) refreshStatus() ใช้แสดงผลในส่วนของหน้าต่าง Controls

7) toggle() เปิดปิดการทำงานของโมดูล

8) handleDependencies() ตรวจสอบว่ามีเครื่องมือ ngrep ติดตั้งไว้หรือไม่

9) handleDependenciesStatus() ตรวจสอบว่ากำลังติดตั้ง dependencies อยู่หรือไม่

10) refreshOutput() ใช้รีเฟรชผลลัพธ์ของการรันคำสั่ง

11) refreshHistory() ใช้รีเฟรชรายชื่อไฟล์ประวัติการทงาน

12) viewHistory() ใช้ดูไฟล์ประวัติการทำงาน

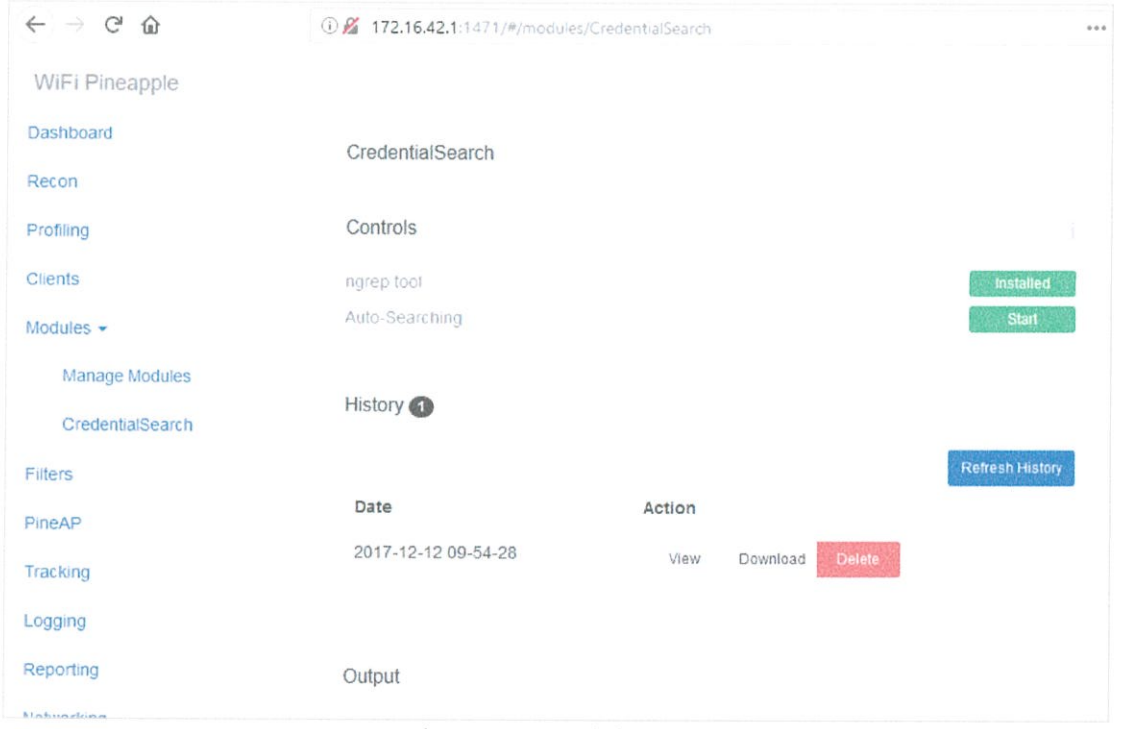
13) deleteHistory() ใช้ลบไฟล์ประวัติการทำงาน

14) downloadHistory() ใช้ดาวน์โหลดไฟล์ประวัติการทำงาน

บทที่ 4
ผลการวิจัย

4.1 ผลการติดตั้งโมดูล

หลังจากที่อัพโหลดไฟล์ทั้งหมดที่เกี่ยวข้องกับโมดูล ขึ้นไปยังที่ตั้ง /pineapple/modules/ บน WiFi Pineapple เรียบร้อยแล้ว ให้รีเฟรช Web interface หนึ่งรอบ จะสังเกตเห็นว่า มีโมดูลใหม่ชื่อ CredentialSearch เพิ่มเข้ามาในเมนู Modules ดังรูป

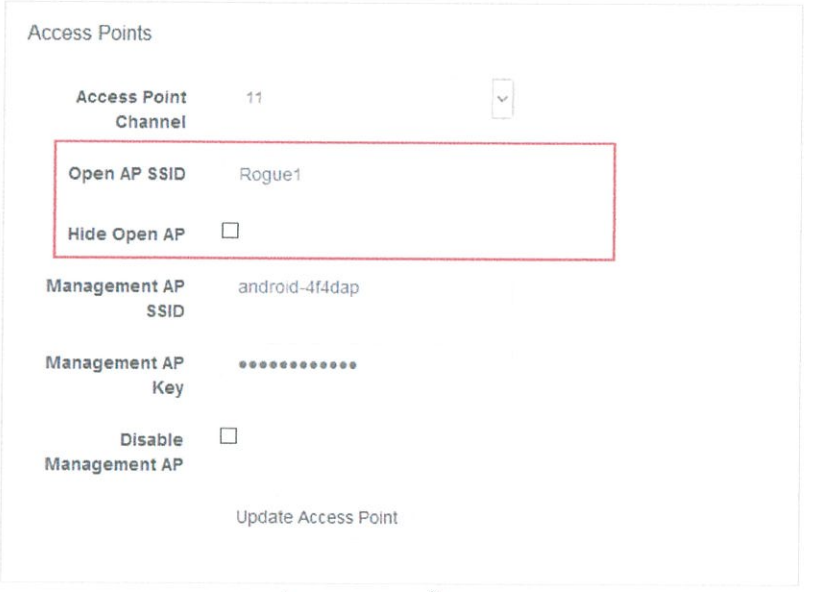


ภาพที่ 4.1 แสดงโมดูลที่เพิ่มเข้ามาใหม่

ถือว่าการติดตั้งโมดูลเสร็จสมบูรณ์ พร้อมใช้งานแล้ว

4.2 ผลการใช้งานโมดูล

หลังจากติดตั้ง Module เสร็จเรียบร้อย ขั้นต่อมาคือการสร้างสัญญาณ Wireless ขึ้นมา โดยการไปที่เมนู Networking ให้นำเครื่องหมายถูกในช่อง Hide open AP ออก และตั้งชื่อสัญญาณในช่อง Open AP SSID ในที่นี้ตั้งชื่อว่า Rogue1 เสร็จแล้วให้กด Update Access Point



ภาพที่ 4.2 แสดงการตั้งค่า Fake AP

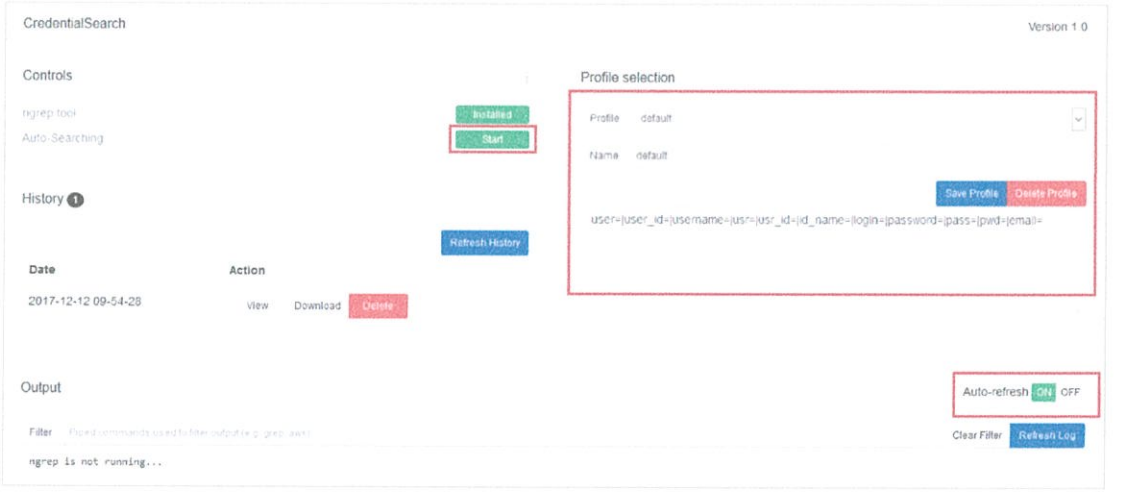
ในส่วนของเครื่อง Client device ให้รอซักครู่ จะพบว่ามีสัญญาณ Wireless ชื่อ Rogue1 โผล่ขึ้นมา

ภาพที่ 4.3 Fake AP ที่แสดงบน Client device

เมื่อทดลองปล่อยสัญญาณ Wireless และให้ Client device เชื่อมต่อเข้ามา สังเกตได้ว่าในหน้า Dashboard จะขึ้นว่า มี 1 currents connected

ภาพที่ 4.4 หน้า Dashboardแสดงจำนวน Devices

หลังจากนั้น เข้าไปที่หน้าโมดูลที่สร้างขึ้นมา ใส่คำค้นที่เราต้องการ หรืออาจจะเลือกจากโปรไฟล์ และกดปุ่ม Start เพื่อเริ่มคัดกรองข้อมูล



ภาพที่ 4.5 Web interface ของโมดูลที่พร้อมใช้งาน

ในส่วนของหน้าต่าง Output อาจจะพิจารณาเปิด Auto-refresh เพื่อให้เห็นข้อมูลที่วิ่งอยู่แบบเรียลไทม์

Output

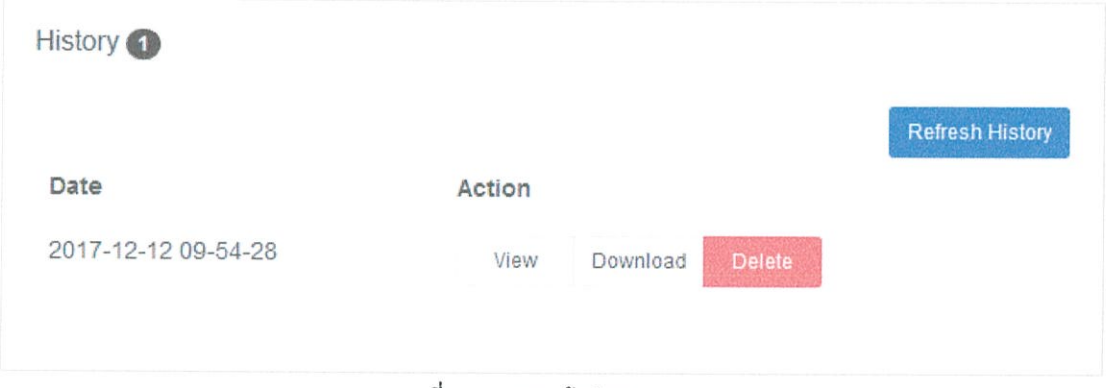
Filter Piped commands used to filter output (e.g. grep, awk)

```
Connection: keep-alive.
Content-Length: 110.
Cache-Control: max-age=0.
Origin: http://188.166.177.132.
Upgrade-Insecure-Requests: 1.
Content-Type: application/x-www-form-urlencoded.
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/62.0.3202.84 Safari/5
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8.
Referer: http://188.166.177.132/wp-login.php.
Accept-Encoding: gzip, deflate.
Accept-Language: en-US,en;q=0.9,th;q=0.8.
Cookie: wp-settings-1=mfold%3Do%26hidetb%3D1%26editor%3Dtinymce%26libraryContent%3Dbrowse%26advImgDetails%3Dsho
.
log=username&pwd=password&wp-submit=Log+In&redirect_to=http%3A%2F%2F188.166.177.132%2Fwp-admin%2F&testcookie=1
##########################################################################################################################
```

ภาพที่ 4.6 ผลลัพธ์ของ Packets ที่กรองออกมา

ในส่วนของ POST Data จะเห็นมีข้อมูลที่สำคัญ คือ log=username&pwd=password ซึ่งตรงกับคำค้นที่เลือกเอาไว้จากโปรไฟล์

ในส่วนของหน้าต่าง History จะพบว่ามีประวัติการเปิดใช้งาน 1 ครั้ง และสามารถกด View เพื่อดูย้อนหลัง หรือกด Download เพื่อบันทึกไฟล์ และกด Delete เพื่อลบประวัติทิ้งได้



ภาพที่ 4.7 แสดงหน้าต่าง History

จะเห็นได้ว่า โมดูลที่สร้างขึ้นนี้ช่วยลดระยะเวลาที่ใช้ในการทดสอบความปลอดภัยได้ และยังช่วยเพิ่มความสะดวกต่อการใช้งานผ่าน Web interface อีกด้วย

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลของโครงงาน

หลังจากการทดลองสร้างโมดูลเพื่อติดตั้งบน WiFi Pineapple TETRA ในส่วนของการแสดงผลผ่าน Web interface นั้นถ้าผลลัพธ์มีปริมาณมาก จะทำให้การแสดงผลไม่เป็นไปตามที่คาดหวัง ในส่วนการทำงานเบื้องหลังของเครื่องมือ ngrep นั้นทำงานได้ดี ผลลัพธ์ของการกรองข้อความนั้น ถูกเก็บไว้ในไฟล์โดยไม่มีปัญหา และสามารถนำไปใช้งานในขั้นตอนต่อไปของการทดสอบเจาะระบบได้เป็นอย่างดี

ปัญหาที่พบระหว่างการพัฒนา คือตอนที่ผู้จัดทำเข้ามาทำโครงงานนี้ ต้องใช้ภาษา JavaScript, Bootstrap framework และ AngularJS framework ซึ่งผู้จัดทำไม่มีความชำนาญในภาษาเหล่านี้ จึงทำให้การทำงานในช่วงแรกนั้นล่าช้า และการทำโครงงานนี้ เป็นการพัฒนาโมดูลที่อยู่บนแพลตฟอร์ม ซึ่งเป็นการเพิ่มฟังก์ชันเข้าไปใน ซึ่งเป็นโค้ดที่มีการทำไว้ก่อนแล้ว ทำให้ต้องทำความเข้าใจการทำงานของระบบก่อน ก่อนที่จะลงมือเขียนโปรแกรม ซึ่งใช้เวลาพอสมควร

5.2 ข้อเสนอแนะ

5.2.1 การพัฒนาต่อ

ณ ขณะนี้ โมดูลสามารถกรองรูปแบบของข้อความที่กำหนด ได้จาก Network interface เพียงอย่างเดียว จึงเป็นการทำงานแบบ Stand alone ไม่ได้เกี่ยวข้องกับโมดูลอื่น ๆ ในขณะที่โมดูล SSLSplit ซึ่งมีความสามารถอ่านข้อความบนโปรโตคอล HTTPS ได้ ถ้าสามารถผนวกความสามารถของโมดูล SSLSplit เข้ากับโมดูลที่ผู้จัดทำพัฒนาขึ้นมา จะสามารถกรองข้อความที่วิ่งบนโปรโตคอล HTTPS ได้ ไม่จำกัดอยู่เพียงแค่ HTTP

5.2.2 รูปแบบ และความซ้ำซ้อน

การเขียนโค้ด ควรจะเป็นไปในรูปแบบ ที่นักพัฒนาโมดูลคนอื่น ๆ เขียน เนื่องจากมีมีคนมาพัฒนาต่อ จะได้เข้าใจได้ง่าย และควรทำการตรวจสอบโค้ดที่เขียน ว่าไม่มีการซ้ำซ้อน หรือเขียนโค้ดโดยไม่จำเป็น

เอกสารอ้างอิง

WiFi Pineapple. 2017. **WiFi Pineapple - Home.** [Online].
Available : https://www.wifipineapple.com/

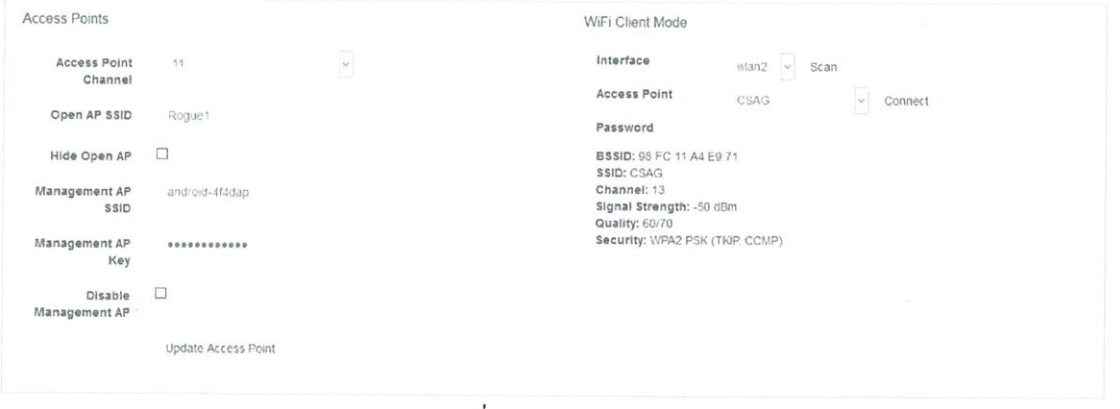
WiFi Pineapple. 2017. **WiFi Pineapple WiKi.** [Online].
Available : https://wifipineapple.github.io/wifipineapple-wiki//#

Wikipedia. 2017. **OpenWrt - Wikipedia.** [Online].
Available : https://en.wikipedia.org/wiki/OpenWrt

Tecmint. 2017. **10 SCP Commands to Transfer Files/Folders in Linux.**
[Online].Available : https://www.tecmint.com/scp-commands-examples/

ภาคผนวก ก
การตั้งค่า WiFi Pineapple เบื้องต้น

เข้าไปที่ http://172.16.42.1:1471 แล้วคลิกที่เมนู Network จะพบกับหน้าต่างดังภาพข้างล่างนี้

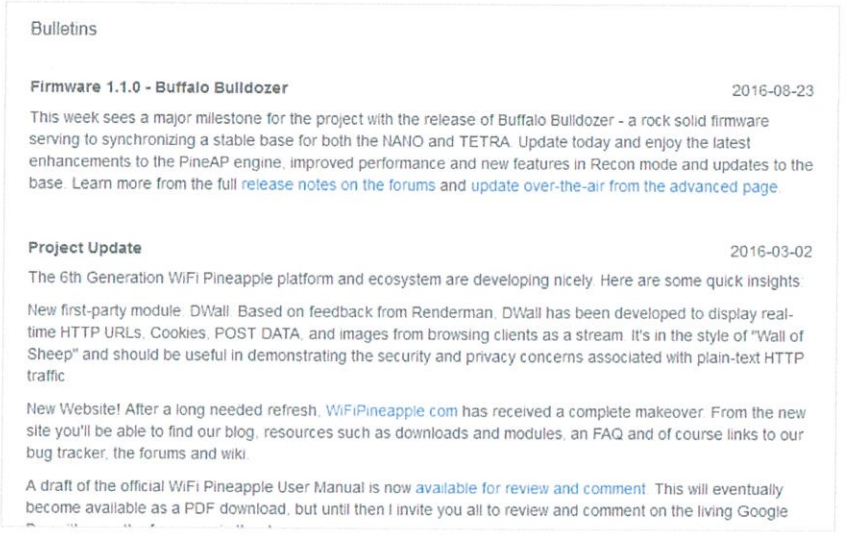


ภาพที่ ก.1 เมนู Network

ในกรอบ WiFi Client Mode ให้ทำการเชื่อมต่อ WiFi Pineapple เข้ากับ Access Point อื่น เช่น เปิด Mobile Hotspot ให้กับ WiFi Pineapple เพื่อให้ Client devices ที่เชื่อมต่อกับ WiFi Pineapple สามารถออกสู่อินเตอร์เน็ตได้

ในกรอบ Access Points ให้กำหนดชื่อของ Open Access Point และนำเครื่องหมายหน้า Hide Open AP ออก เพื่อกระจายสัญญาณ Wireless ที่สร้างขึ้นมา ส่วน Management AP หมายถึง Access Point อีกชื่อหนึ่ง มีไว้สำหรับเข้าจัดการหน้า WiFi Pineapple ความแตกต่างก็คือ ทั้งสอง Access Point จะใช้ Network คนละ Interface กัน

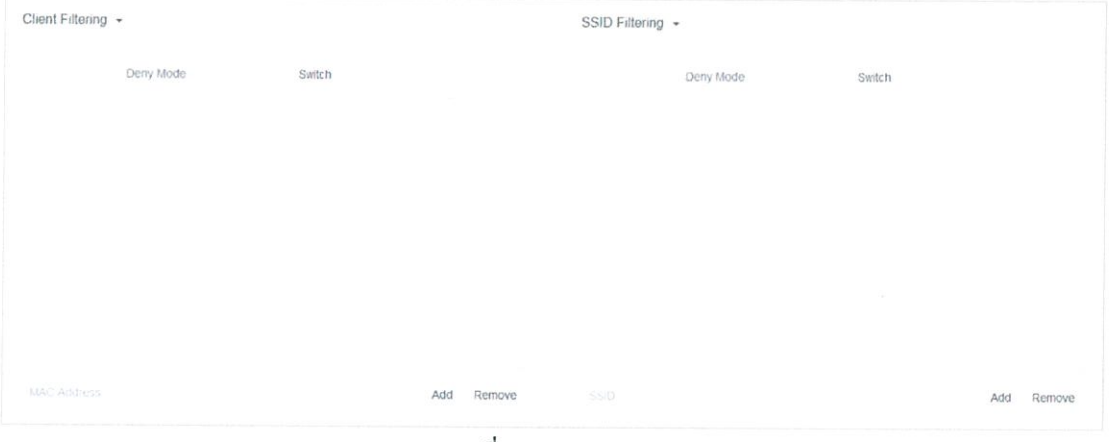
วิธีการตรวจสอบว่า WiFi Pineapple ได้เชื่อมต่ออินเตอร์เน็ตอยู่หรือไม่ ให้เลือกเมนู Dashboard ในส่วนหน้าต่าง Bulletins ให้คลิก Load Bulletins from WiFiPineapple.com ถ้าปรากฏข้อความดังภาพด้านล่าง หมายความว่า WiFi Pineapple ได้เชื่อมต่ออินเตอร์เน็ตเรียบร้อยแล้ว



ภาพที่ ก.2 ตรวจสอบการเชื่อมต่อจาก Bulletins

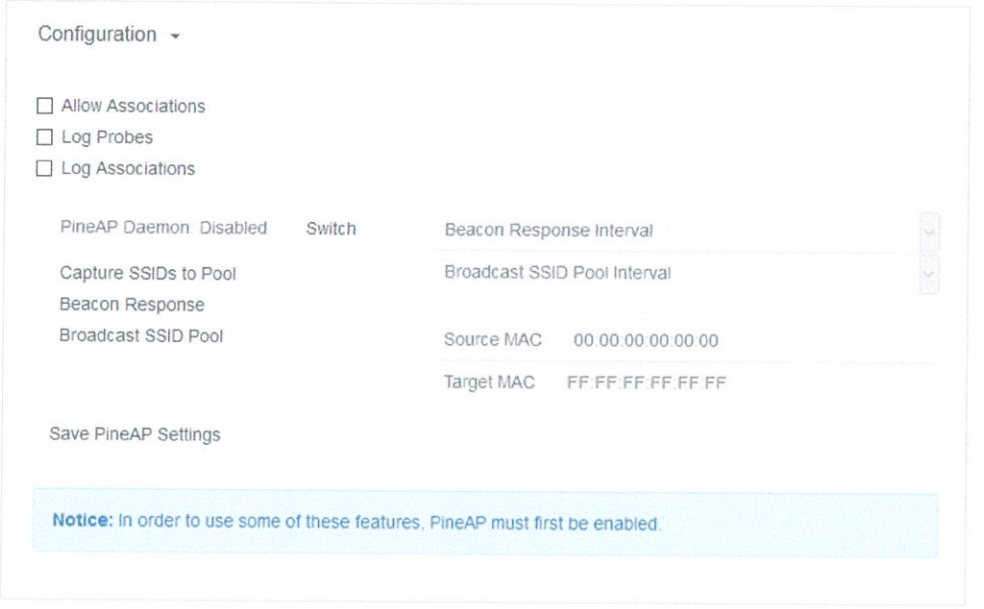
ขั้นตอนถัดมา ให้ดาวน์โหลดโมดูลที่จะใช้งาน โดยการเลือกเมนู Modules > Manage Modules คลิกที่ Get Modules from WiFiPineapple.com หลังจากนั้นให้เลือกโมดูลที่ต้องการ แล้วกด Install

เมนู Filters ใช้กำหนดค่าว่าจะอนุญาตให้ Client devices ที่มี MAC address ใดบ้างเชื่อมต่อกับ WiFi Pineapple ได้ และจะอนุญาตให้ WiFi Pineapple กระจาย SSID ชื่อใดออกไปได้บ้าง



ภาพที่ ก.3 เมนู Filters

สุดท้ายในส่วนของเมนู PineAP มีการตั้งค่าที่น่าสนใจดังนี้



ภาพที่ ก.4 เมนู PineAP

- Allow Associations

อนุญาตให้ Client Devices เชื่อมต่อ Access Point ชื่ออื่นนอกเหนือจาก Open AP ที่ได้ตั้งค่าไว้ในเมนู Network

- Log Probes

เก็บบันทึก Probe request ของ Client devices

- Log Associations

เก็บบันทึก เมื่อ Client devices ทำการเชื่อมต่อกับ Access Point ใด ๆ

- Capture SSIDs to Pool

บันทึกทุก SSID ที่ดักจับไว้ได้ โดยบันทึกไว้ใน Pool (มีจุดประสงค์เพื่อนำชื่อ SSID มาใช้งานภายหลัง)

- Beacon Response

เมื่อ Client devices ส่ง Probe request ใด ๆ ออกมาว่ามี Access Point ที่ร้องขออยู่บริเวณโดยรอบหรือไม่ จะอนุญาตให้ WiFi Pineapple ตอบกลับไปว่าตัว WiFi Pineapple เองคือ Access Point ที่ Client ร้องขอ

- Broadcast SSID Pool

อนุญาตให้ WiFi Pineapple กระจายสัญญาณ Wireless ด้วย SSID ที่อยู่ใน Pool ทั้งหมด