Single Point Authentication by Multiple Factor Authentication
การยืนยันตัวตน ณ จุดเดียวโดยใช้พหุปัจจัย

ปรัชญา ไชยเมือง[1], สมนึก พ่วงพรพิทักษ์[2] และ วิรัตน์ พงษ์ศิริ[3]

คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม

41/20 ตำบลขามเรียง อำเภอกันทรวิชัย จังหวัดมหาสารคาม 44150 โทรศัพท์: 0-4375-4322 ต่อ 2414

pratchaya.cmg@gmail.com[1], somnuk.p@msu.ac.th[2], wirat@msu.ac.th[3]

ABSTRACT – Nowadays, Single Point Authentication is very a significant to solve a distributed authentication process. Due to the Single Sign on (SS0) by LDAP is popular solution. However, it still has some security vulnerabilities because a password is also vulnerable to leaking and forgetful. Although, There have been several solutions, such as TFA and OTP proposed previously to enhance the password leaking and forgetfulness but, they still have a lot of drawbacks. This paper analyzes the drawbacks of the previous solutions; then design and implement a new solution using LDAP, Web Services, Java program and J2ME technologies. By designing the authentication process at a single point authenticator and using multi-factor authentication together with challenge-response new S/Key OTP techniques. The new design called "Single Point Authentication by Multiple Factors Authentication (SPA-MFA)". From experiments/evaluation, SPA-MFA is very effective, low cost, cheap and easy to implement.

KEYWORDS – Single Point Authentication, Multiple Point Authentication, One Time Password, Password Recovery

บทคัดย่อ -- ในปัจจุบัน การยืนยันตัวตนผู้ใช้ ณ จุดเดียว เป็นวิธีการที่สำคัญมากในการแก้ปัญหาการกระจายของกระบวนการยืนยันตัวตนผู้ใช้ในระบบสารสนเทศ และ Single Sign-on โดย LDAP ได้ถูกนำเสนอให้เป็นหนึ่งในการแก้ปัญหาที่นิยมใช้งานอย่างแพร่หลาย แต่อย่างไรก็ตามการใช้รหัสผ่านอย่างเดียวเพื่อยืนยันตัวตนใน LDAP มีความเสี่ยงเช่น การรั่วไหลของรหัสผ่าน การลืมรหัสผ่าน ถึงแม้ว่ามีข้อเสนอ TFA และ OTP เพื่อแก้ปัญหาการรั่วไหล แต่ยังพบปัญหาที่ตามมาจากข้อเสนอทั้งสอง อาทิเช่น ความไม่เหมาะสมในการเลือกใช้อุปกรณ์ ต้นทุนการใช้งานสูง มีความเสี่ยงจาก Brute Force Attack เป็นต้น และข้อเสนอเพื่อแก้ปัญหาการลืมรหัสผ่าน เช่น คำถามกันลืม กู้คืนรหัสผ่านด้วยอีเมล ยังมีจุดอ่อนในหลายประการ ในงานวิจัยนี้จึงได้ออกแบบพร้อมพัฒนาโปรแกรมต้นแบบการแก้ปัญหาใหม่ เพื่อทำให้กระบวนการยืนยันตัวตนดีขึ้น โดยการออกแบบให้กระบวนการยืนยันตัวตนเกิดขึ้น ณ จุดเดียว และใช้การยืนยันตัวตนแบบพหุปัจจัยร่วมกับเทคนิค OTP แบบใหม่ ทำให้แนวคิดในการแก้ปัญหาแบบใหม่นี้มีประสิทธิผลสูงกว่า มีต้นทุนที่ถูก และง่ายต่อการติดตั้งใช้งาน

คำสำคัญ -- การยืนยันตัวตน ณ จุดเดียว, การยืนยันตัวต้นด้วยพหุปัจจัย, รหัสผ่านแบบใช้ครั้งเดียว, การกู้คืนรหัสผ่าน

1. บทนำ

ระบบสารสนเทศส่วนใหญ่ต้องการยืนยันตัวผู้ใช้เพื่อเข้าใช้ระบบ (Login) โดยใช้ปัจจัยการยืนยันตัวตนแบบต่างๆ แต่ที่นิยมมากสุดคือใช้ รหัสผ่าน ซึ่งส่วนใหญ่หนึ่งระบบใช้หนึ่งรหัสผ่าน ทำให้ผู้ใช้มีจำนวนรหัสผ่านมากขึ้นหากใช้งานหลายๆ ระบบ หรือเรียกว่า Multiple Point Authentication จึงได้มีข้อเสนอหนึ่งเพื่อแก้ปัญหาดังกล่าว คือ Single Sign-On (SSO) [19]โดยมีแนวคิดให้ใช้รหัสผ่านเดียว Login ได้หลายๆ ระบบ ซึ่งการทำ SSO ส่วนใหญ่ในปัจจุบันนิยมใช้มาตรฐาน LDAP [26] ในการจัดเก็บข้อมูลเพื่อใช้ตรวจสอบ แต่การใช้ LDAP มีจุดอ่อนคือ กระบวนการยืนยันตัวตนผู้ใช้ใน LDAP เลือกใช้รหัสผ่านซึ่งเป็น User Knowledge [15] เพียงอย่างเดียว ซึ่งรหัสผ่านมีจุดอ่อนคือ รั่วไหลได้หลายวิธี (ถูกดักจับข้อมูลผู้ใช้แพร่งพราย เขียนไว้กันลืม) และการลืมรหัสผ่าน ทำให้การใช้ LDAP สำหรับทำ SSO มีความเสี่ยงเมื่อรหัสผ่านรั่วไหลและเมื่อลืมรหัสผ่าน

จากการศึกษาวิธีแก้ปัญหาการรั่วไหลของรหัสผ่าน พบว่า การยืนยันตัวด้วยปัจจัยสองแบบ (Two Factor Authentication หรือ TFA) [14] และ รหัสผ่านแบบใช้ครั้งเดียว (One Time Password หรือ OTP) [18] เป็นตัวอย่างข้อเสนอเพื่อแก้ปัญหาเรื่องรหัสผ่านรั่วไหลที่นิยมใช้งานในปัจจุบัน โดยมีแนวคิดให้ใช้ User Possession [15] ที่ผู้ใช้ถือครองอยู่เข้ามาเป็นอีกปัจจัยหนึ่งในการยืนยันตัวตน และวิธีการที่นิยมใช้มากที่สุดคือ ใช้ User Possession แสดง OTP ซึ่งเปลี่ยนแปลงทุกครั้งเมื่อ Login แต่ยังพบจุดอ่อนหลายประการที่เกิดจากข้อเสนอ TFA เช่น ความไม่เหมาะสมของการเลือกใช้งานอุปกรณ์ การติดตั้งระบบเพื่อใช้งานยุ่งยาก และต้นทุนในการใช้งานสูง เช่นเดียวกันกับจุดอ่อนของ OTP คือ ความเสี่ยงในการ Brute Force Attack [16], การทำงานที่ไม่ผสานงานกันระหว่าง Server กับ Client, จำนวนตัวอักษรของ OTP มีมาก เป็นต้น นอกจากนั้นได้ศึกษาการแก้ปัญหาการลืมรหัสผ่าน พบว่ามีข้อเสนอสามแบบที่นิยมใช้ คือ คำถามกันลืม, กู้คืนรหัสผ่านด้วยอีเมล, และกู้คืนรหัสผ่านด้วยผู้แลระบบ ซึ่งวิธีการต่างๆ เหล่านี้ล้วนแล้วแต่มีจุดอ่อน เช่น คำถามกันลืมคาดเดาได้ง่าย ลืมคำถามกันลืม ระบบอีเมลถูกแทรกแซงการสื่อสารได้ ผู้ดูแลระบบมีภาระมากขึ้น เป็นต้น

งานวิจัยนี้จึงได้นำเสนอแนวคิดในการแก้ปัญหาใหม่ โดยออกแบบการยืนยันตัวตนผู้ใช้ไว้ ณ จุดเดียว ด้วย LDAP ซึ่งแบ่งกระบวนการยืนยันตัวตนผู้ใช้ออกเป็น 2 ส่วน คือการยืนยันตัวตนเพื่อเข้าใช้ระบบ (User Login) ด้วยปัจจัย 2 แบบคือ User Knowledge เลือกใช้รหัสผ่าน ร่วมกับ User Possession เลือกใช้โทรศัพท์มือถือที่ผู้ใช้พกพาอยู่แล้ว โดยการออกแบบพัฒนาโปรแกรม J2ME เพื่อสร้าง OTP ที่มีความเสี่ยงจากการ Brute Force Attack น้อยที่สุดคือ S/Key พร้อมกับลดจำนวนตัวอักษร OTP ของ S/Key [23] ให้น้อยลงกว่าเดิม และส่วนที่ 2 คือการยืนยันตัวเพื่อกู้คืนรหัสผ่าน (User Recover Password) โดย User Attribute [15] ในรูปของลายนิ้วมือผู้ใช้ โดยออกแบบและพัฒนาโปรแกรมเพื่อติดต่อกับเครื่องอ่านนิ้วมือและ LDAP Server เพื่ออ่านค่าและตรวจสอบลายนิ้วมือในกระบวนการดังกล่าว นอกจากนี้ได้ออกแบบให้ใช้โทรศัพท์มือถือ เพื่อกู้คืนรหัสผ่านด้วย SMS ที่ส่งรหัสผ่านใหม่จากโทรศัพท์มือถือของผู้ใช้ผ่าน GSM Modem ที่ติดตั้งอยู่กับ Server ของระบบ ทำให้มีระบบกู้คืนรหัสผ่านด้วย User Possession ซึ่งเป็นอีกช่องทางหนึ่งในกระบวนการนี้ นอกจากนี้ยังได้ออกแบบและพัฒนาระบบต้นแบบ และเรียกว่า Single Point Authentication by Multiple Factor Authentication (SPA-MFA) ซึ่งเป็นระบบที่สามารถแก้ปัญหาต่างๆ ที่เกิดจากการใช้รหัสผ่านเพียงอย่างเดียว มีความมั่นคง ถูกต้อง และน่าเชื่อถือ รวมถึงมีการทดสอบระบบในการแก้ปัญหา และสรุปผลการทดลอง

ในส่วนที่ 2 ของเอกสารฉบับนี้ได้กล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้อง สำหรับส่วนที่ 3 กล่าวถึง การออกแบบและพัฒนาระบบ และในส่วนที่ 4 และ 5 กล่าวถึง ผลการพัฒนาโปรแกรมและวิเคราะห์ผลการพัฒนาตามลำดับ สำหรับบทสรุปจะอยู่ในส่วนที่ 6

2. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 Lightweight Directory Access Protocol (LDAP)

LDAP [26] เป็น Protocol X.509 ที่พัฒนามาจาก Protocol X.500 เพื่อใช้ Access และ Update ข้อมูลของ Directory ซึ่ง Directory อาจเรียกได้ว่าเป็นฐานข้อมูลแบบพิเศษ โดยคุณสมบัติที่โดดเด่นของ LDAP คือ มีความเร็วในการเข้าถึงข้อ

มูลสูง และทำให้แอปพลิเคชั่นที่ทำงานบนโพรโทคอลเหล่านี้สามารถเข้าถึงข้อมูลได้อย่างรวดเร็ว ทำให้ LDAP เป็นที่ยอมรับและนำมาใช้งานทั่วไปในกระบวนการยืนยันตัวตน จากการศึกษา LDAP พบว่าการยืนยันตัวตนผู้ใช้ใน LDAP เลือกใช้รหัสผ่านเพียงอย่างเดียว ซึ่งการใช้รหัสผ่านเพียงอย่างเดียวนั้นมีจุดอ่อนหลายประการคือ รหัสผ่านรั่วไหลได้หลายวิธี เช่น ถูกดักจับข้อมูล ผู้ใช้บอกต่อผู้อื่นโดยไม่ตั้งใจ ใช้รหัสผ่านที่คาดเดาง่าย เป็นต้น เมื่อรหัสผ่านรั่วไหลไป จะถูกนำไปใช้ Login ซ้ำได้ด้วยรหัสผ่านตัวเดิมตลอด โดยระบบและผู้ใช้เองไม่สามารถตรวจสอบได้เลยว่ารหัสผ่านนั้นถูกใช้จากผู้ใช้ที่ได้รับอนุญาตหรือไม่ จุดอ่อนอีกประการ คือผู้ใช้ลืมรหัสผ่าน และเมื่อลืมรหัสผ่านไปแล้วผู้ใช้จะไม่สามารถเข้าใช้งานระบบได้อีกต่อไป แต่ด้วยความสามรถในเรื่องความเร็วในการเข้าถึงข้อมูลของ LDAP และการแน่น Access ข้อมูลเพียงอย่างเดียว LDAP จึงเหมาะสมที่จะนำมาใช้ในกระบวนการยืนยันตัวตน เพราะการยืนยันตัวตนส่วนใหญ่ทำโดยดึงข้อมูลที่เก็บไว้ขึ้นมาตรวจสอบกับข้อมูลที่ส่งให้ระบบ มากกว่าการ Update ข้อมูล ดังนั้นงานวิจัยนี้จึงศึกษาการทำงานของ LDAP เพื่อใช้ความสามารถที่โดดเด่นของ LDAP มาใช้ในกระบวนการยืนยันตัวตนที่ออกแบบใหม่นี้ พร้อมกับแก้ไขจุดอ่อนที่เกิดจากการใช้รหัสผ่านเพียงอย่างเดียวใน LDAP โดยใช้ TFA และ OTP ซึ่งจะได้กล่าวในส่วนถัดไป

2.2 Two Factor Authentications (TFA)

TFA [14] คือกระบวนการยืนยันตัวตนโดยใช้ปัจจัยสองอย่างร่วมกันซึ่งวิธีการที่ได้รับความนิยมใช้งานอย่างแพร่หลายมากที่สุดคือ ใช้ User knowledge ในรูปของรหัสผ่าน ร่วมกับ User Possession ในรูปของ Security Token หรือ อุปกรณ์เสริมเพื่อเพิ่มความมั่นคง ทำให้การยืนยันตัวตนมีความปลอดภัยมากขึ้น เนื่องจากจำนวนปัจจัยที่เพิ่มมากขึ้นส่งผลให้กระบวนการยืนยันตัวตนมีความซับซ้อนและมีลำดับขั้นตอนเพิ่มมากขึ้น จากงานวิจัยและผลิตภัณฑ์ เช่น Aradiom SolidPass [1], AuthAnvil [2], FireID [3], FiveBarGate [4], Smart Card Authentication [20], RSA SecurID [12] , Diversinet [5] เป็นตัวอย่างข้อเสนอการใช้งาน TFA แต่การใช้งาน TFA ในข้อเสนอเหล่านั้นยังมีจุดอ่อนหลายประการเช่น ความไม่เหมาะสมกับการเลือกใช้อุปกรณ์ ความไม่เป็นอิสระต่อระบบปฏิบัติการ ความยุ่งยากและข้อจำกัดในการติดตั้งระบบ ปัญหาเรื่องค่าใช้จ่ายในการติดตั้งใช้งานระบบทั้งเรื่องลิขสิทธิ์ซอฟต์แวร์ และเรื่องค่าใช้จ่ายกับ Security Token เป็นต้น งานวิจัยนี้จึงได้ใช้แนวคิดของ TFA มาปรับปรุงใช้ ในการแก้ปัญหาการรั่วไหลของรหัสผ่าน และมุ่งเน้นแก้ปัญหาที่เกิดขึ้นจากการใช้ TFA แบบเดิม

2.3 One Time Password (OTP)

OTP [18] หรือรหัสผ่านแบบใช้ครั้งเดียว คือ วิธีการหนึ่งที่แก้ปัญหาการใช้รหัสผ่าน ซึ่งมีแนวคิดให้รหัสผ่านที่ใช้ Login เปลี่ยนแปลงอยู่ตลอดทุกครั้ง ส่วนใหญ่ใช้ OTP เพื่อทำ TFA วิธีการคือให้ User Possession ที่ผู้ใช้ถือครองอยู่แสดง OTP เพื่อเป็นรหัสผ่านส่วนที่สองในการ Login ในปัจจุบันมีมาตรฐานที่นิยมใช้งานกัน 3 แบบดังนี้ HOTP [17], TOTP [25] และ S/Key โดย OTP ทั้งสามแบบมีการแสดง Output แตกต่างกัน การศึกษา OTP ในงานวิจัยนี้เพื่อวิเคราะห์ความสามารถการป้องกันการโจมตีระบบด้วยการเดาสุ่มรหัสผ่านทั้งหมดที่เป็นไปได้ (Brute Force Attract Password) ซึ่ง OTP แต่ละแบบมีความสามารถนี้ต่างกัน โดยดูจากความเป็นได้ทั้งหมดของการแสดง Output ของ OTP แต่ละแบบ ดังต่อไปนี้ HOTP และ TOTP แสดงผลเป็นตัวเลขทั้งหมด 8 หลัก ความเป็นไปได้ที่จะเกิด Output ทั้งหมดคือ 108 ส่วน S/Key แสดงคำที่กำหนดไว้ 6 คำ จากคำทั้งหมด 2048 คำ(อธิบายการทำงานไว้ใน Appendix B ของ RFC 2289 [23]) ความเป็นไปได้ที่จะแสดง Output ทั้งหมด คือ 20486 หรือประมาณ 1019 จากความเป็นได้ของ Output ทั้งหมดเห็นได้ว่า S/Key มีมากกว่าแบบ HOTP และ TOTP ทำให้การ Brute Force Attract ทำได้ยากกว่า แต่จุดอ่อนของ S/Key คือ จำนวนตัวอักษรที่แสดงมีได้มาถึง 24 ตัว และในงานวิจัยนี้ได้ทำการศึกษาและวิเคราะห์ความเสี่ยงจาก Brute Force Attract ใน OTP แต่ละประเภทและนำ OTP แบบที่เหมาะสมไปใช้งานกับระบบที่ออกแบบใหม่เพื่อทำ TFA พร้อมกับหาแนวทางในการแก้ไขจุดอ่อนที่เกิดขึ้นใน OTP แบบนั้นๆ

2.4 Single Sign on (SSO)

SSO [19] คือ การควบคุมการเข้าสู่ระบบต่างๆ หลายระบบ ได้ด้วยการยืนยันตัวตน ณ จุดเดียว ซึ่งเป็นเทคโนโลยีที่อำนวยความสะดวกต่อการ Login ในสารสนเทศหลายๆ ระบบ ได้ด้วยรหัสผ่านเพียงตัวเดียว ซึ่งในปัจจุบัน SSO ได้ถูกพัฒนาใช้งาน

ในระบบสารสนเทศชั้นนำของโลกหลายแห่ง เช่น Google.com และ Facebook.com มีการพัฒนา API ที่สามารถช่วยให้ระบบสารสนเทศต่าง Login ด้วยบัญชีผู้ใช้ ของ Google หรือ Facebook ได้ เช่น Google Single Sign on API และ Facebook Single Sign on เป็นต้น การศึกษา SSO ในงานวิจัยนี้เพื่อออกแบบระบบการยืนยันตัวตนใหม่ในลักษณะ SSO เพื่อแก้ปัญหา Multi Point Authentication ที่เกิดขึ้นจากการใช้รหัสผ่านในการยืนยันตัวตน แต่ SSO ที่นิยมใช้งานกันยังมีจุดอ่อนในเรื่อง การกู้คืนรหัสผ่านที่ยังไม่เหมาะสมไม่ว่าจะเป็น Google หรือ Facebook ยังใช้คำถามกันลืม และ ใช้ E-Mail Recover Password เป็นต้น ซึ่งจุดอ่อนในกระบวนดังกล่าวนี้จะได้กล่าวในส่วนถัดไป

2.5 Password Recovery

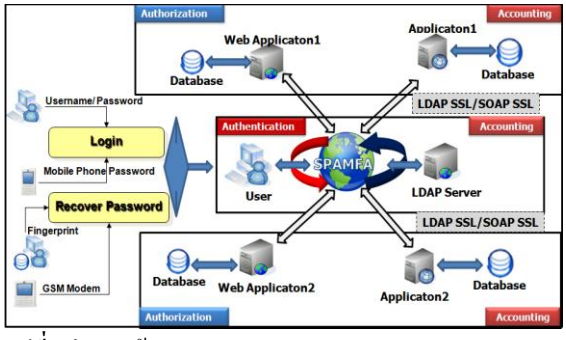
การกู้คืนรหัสผ่าน (Password Recovery) [24] คือวิธีการที่ใช้แก้ไขปัญหาการลืมรหัสผ่าน ซึ่งในปัจจุบันมี 3 วิธีที่ได้รับความนิยมใช้งาน ได้แก่ (1) ใช้คำถามกันลืม คือการตั้งคำถามและคำตอบพิเศษ สำหรับเป็นข้อมูลในการกู้คืนรหัสผ่าน แต่คำถามกันลืมเองมีจุดอ่อน คือการตั้งคำถามและคำตอบง่ายต่อการคาดเดา ผู้ใช้ลืมคำถามกันลืมที่ตั้งไว้ (2) E-Mail Recover Password คือการส่งรหัสผ่านใหม่ผ่านทางอีเมลในกรณีลืมรหัสผ่าน ซึ่งอีเมลนั้นมีจุดอ่อนดังที่กล่าวไว้ในหัวข้อ 2.3 และ (3) กู้คืนรหัสผ่านโดยผู้ดูแลระบบ คือ ผู้ดูแลระบบเป็นผู้กู้คืนรหัสผ่านให้กับผู้ใช้ จุดอ่อนที่เกิดขึ้นคือผู้ดูแลระบบต้องมีภาระเพิ่มขึ้น ในส่วนนี้หากระบบมีผู้ใช้จำนวนมากๆ งานวิจัยนี้ได้ศึกษาวิธีการกู้คืนรหัสผ่านเพื่อหาแนวทางในการแก้ปัญหาที่เกิดขึ้นจากวิธีการที่ได้นำเสนอมาก่อนหน้างานนี้ เพื่อทำให้กระบวนการดังกล่าวมีความมั่นคงและน่าเชื่อถือมากขึ้น

3. การออกแบบและพัฒนาระบบ

ได้ทำการออกแบบและพัฒนาระบบโดยให้ชื่อเรียกว่า Single Point Authentication by Multiple Factor Authentication (SAP-MFA) ซึ่งมีรายละเอียดดังแสดงในรูปที่ 1

3.1 ในแง่ประสิทธิผล (Effectiveness)

(1) ออกแบบ Single Point Authentication คือ การนำกระบวนการยืนยันตัวตนของผู้ใช้งานระบบต่างๆ มาไว้ ณ จุดเดียว จะทำให้ผู้ใช้มีเพียงบัญชีผู้ใช้ และรหัสผ่านเดียวเพื่อเข้าใช้งานหลายๆ ระบบได้ ซึ่งแตกต่างจากข้อเสนอ [1], [4], [6], [7], [8], [9] ที่ผู้ใช้ยังต้องมีบัญชีผู้ใช้ที่ต่างกันออกไปตามระบบที่ใช้งานอยู่ และได้ออกแบบให้กระบวนการกำหนดสิทธิ์ (Authorization) การใช้งานระบบเป็นหน้าที่ของระบบนั้นๆ เป็นผู้กำหนดทำให้สามารถลดภาระการกำหนดสิทธิ์ใช้งานระบบให้กับ Server ได้ ซึ่งต่างจากข้อเสนอ [2], [3], [5], [12], [20] ที่ใช้ Server กลางในการยืนยันตัวตนแล้ว Server ต้องรับภาระในการกำหนดสิทธิ์ของผู้ใช้อีกด้วย



รูปที่ 1 โครงสร้างการออกแบบระบบ

(2) ออกแบบการยืนยันตัวต้นเข้าใช้ระบบ (User Login) แบ่งเป็น 3 แบบดังนี้

-ใช้ User Knowledge (Username/Password) เพียงอย่างเดียว คือการ Login แบบปกติ คงไว้ในกรณีที่ระบบเดิมยังต้องการใช้การ Login แบบเดิมหรือผู้ใช้ยังขาด User Possession (โทรศัพท์มือถือ)

-ใช้ User Possession (โทรศัพท์มือถือที่ต้องลงทะเบียนไว้กับระบบก่อนแล้ว) เพียงอย่างเดียว คือรหัสผ่านที่ใช้ Login จะแสดงบนโทรศัพท์มือถือแบบ OTP เรียกว่า “Mobile Phone Password (MPP)”

-ใช้ User Knowledge (รหัสผ่าน) + User Possession (MPP) คือ รหัสผ่านที่ใช้ Login จะประกอบด้วย 2 ส่วน หากมีข้อมูลส่วนใดมีการถูกดักจับหรือรั่วไหลแฮกเกอร์ก็ยังไม่สามารถเข้าสู่ระบบได้ ซึ่งวิธีนี้เป็นวิธีที่ปลอดภัยที่สุด และสามารถแก้ไขปัญหาเรื่องการรั่วไหลของรหัสผ่านได้

(3) การยืนยันตัวตนเพื่อกู้คืนรหัสผ่าน (Recovery Password) ออกแบบในส่วนนี้แบ่งออกเป็น 2 แบบ คือ

- ใช้ User Attribute คือ ลายนิ้วมือ ซึ่งเป็นข้อมูลที่ไม่ลืม ไม่รั่วไหล โดยออกแบบให้มี Fingerprint Software ที่พัฒนา

จาก SDK ที่สามารถใช้ได้กับเครื่องอ่านลายนิ้วมือหลายรุ่น [10] เพื่อจัดเก็บและตรวจสอบข้อมูลลายนิ้วมือ ระหว่างเครื่องอ่านลายนิ้วมือกับ Server นำไปสู่แก้ปัญหาจากการใช้ คำถามกันลืม E-mail Recover Password และการกู้คืนรหัสผ่านด้วยผู้ดูแลระบบ ที่นิยมใช้งานในปัจจุบัน

- ใช้ GSM Modem เพื่อรับ SMS จากผู้ใช้ที่ต้องการกู้คืนรหัสผ่านโดยผู้ใช้กำหนดรหัสผ่านใหม่ด้วยการส่ง SMS จากโทรศัพท์มือถือที่ได้ลงทะเบียนไว้ มาที่ GSM Modem ของระบบ ซึ่งจุดนี้เป็นอีกช่องทางหนึ่งในกรณีที่ระบบนั้น ๆ ยังขาดเครื่องอ่านลายนิ้วมือ

(4) ออกแบบให้มีการเข้ารหัสข้อมูลที่สื่อสาร ระหว่าง SPA-MFA Server กับ Applications ที่เรียกใช้บริการ ด้วย SSL ทั้งการสื่อสารด้วย LDAP และ SOAP เพื่อเพิ่มความมั่นคงให้ข้อมูลมากขึ้นกว่าการส่งข้อมูลที่ไม่ได้ถูกเข้ารหัส

(5) Mobile Phone Password (MPP) คือ OTP บนโทรศัพท์มือถือ ซึ่งเลือกการสร้าง OTP แบบ Challenge Base [23] และแสดงผล Output แบบ S/Key โดยการสร้าง OTP แบบ Challenge Base คือ OTP จะถูกสร้างจากข้อมูล Challenge ที่ Server ส่งให้ใหม่ทุกครั้งที่ Login ทำให้การผสานงานของ OTP ระหว่าง Server กับ Client ไม่ผิดพลาด ซึ่งต่างจากการสร้าง OTP แบบ Counter Base OTP [17] (ใช้ตัวนับที่ตรงกัน) และ Time Base OTP [25] (ใช้เวลาที่ตรงกัน) ที่การผสานงานสร้าง OTP อาจผิดพลาดได้ และการแสดง Output แบบ S/Key นั้นปลอดภัยจาก Brute Force Attract มากที่สุดเพราะความเป็นได้ของ Output มีมากกว่าเมื่อเทียบกับ HOTP และ TOTP นอกจากนั้นยังได้ออกแบบให้ OTP ที่ใช้แสดงผลแบบ BASE64 ได้อีกด้วย ซึ่งผลของ Output แบบ BASE64 แสดงจำนวนตัวอักษรน้อยกว่าของ S/Key คือ 24 แต่ Output แบบ BASE64 แสดงเพียง 11 ตัวอักษร

(6) การออกแบบให้มีการลงทะเบียนโทรศัพท์มือถือทุกๆเครื่องที่ต้องการใช้งานในระบบ โดยการกำหนดรหัสให้กับโทรศัพท์มือถือ (MobileID) เพื่อระบุให้โทรศัพท์มือถือจับคู่ใช้งานกับ Username แบบ 1 ต่อ 1 และการสร้าง MPP ต้องมี MobileID เป็น Seed รวมอยู่ด้วย เมื่อตรวจสอบ MPP จะสามารถทราบได้ว่าถูกสร้างมาจากโทรศัพท์มือถือที่ได้รับอนุญาตหรือไม่ นั้นก็คือ การยืนยันอุปกรณ์ ซึ่งในข้อเสนอก่อนหน้านี้ [2], [4], [7], [8], [9], [17] ยังขาดกระบวนการดังกล่าว

3.2 ในแง่ความง่ายในการนำไปใช้งาน (Feasibility)

(1) การออกแบบ SPA-MFA ให้ใช้มีฟังก์ชั่น ในรูปแบบของ Web Service โดยใช้ Java Web Service เพื่อนำคุณสมบัติของ Java และ Web Service ในเรื่องการติดตั้งและให้บริการได้กับทุกระบบปฏิบัติการ และการพัฒนา Fingerprint Software ด้วย Java และใช้ Finger Print SDK ที่สามารถใช้ ใช้กับเครื่องอ่านลายนิ้วมือได้หลายรุ่นทำให้การติดตั้งใช้งานระบบเป็นอิสระกับทุกๆ ระบบปฏิบัติการ และใช้กับเครื่องอ่านลายนิ้วมือได้หลายรุ่น และเช่นด้วยกันกับการพัฒนาโปรแกรมเพื่อติดต่อกับ GSM Modem ในการรับ SMS ได้พัฒนาด้วย Java เช่นกันดังนั้นการติดตั้งระบบทั้งหมดสามารถทำได้กับทุกระบบปฏิบัติการ ซึ่งต่างจากข้อเสนอเดิม [5], [7], [8], [10] ที่มีปัญหาสามารถใช้งานได้บนระบบปฏิบัติการ Windows เท่านั้น

(2) พัฒนาโปรแกรม J2ME เพื่อติดตั้งที่โทรศัพท์มือถือของผู้ใช้สำหรับใช้สร้าง MPP ทำให้โทรศัพท์มือถือมีความสามารถเป็น Security Token ที่ใช้เป็น User Possession ได้ ดังนั้นจึงไม่จำเป็นต้องมีการลงทุนซื้อหรือพกพา Token อื่นเพิ่มเติม ซึ่งในส่วนนี้เป็นการแก้ปัญหาเรื่อง การเลือกใช้อุปกรณ์ที่เหมาะสมหากเทียบกับงานวิจัย [20], [12] ที่การทำ TFA ยังจำเป็นที่จะต้องลงทุนซื้อและเพิ่มภาระการพกพา Token เช่น Smart Card, RAS SecurID Token เพิ่มเติม

3.3 ในแง่ความสะดวกสบาย (Convenience)

ระบบออกแบบให้ติดตั้ง OpenLDAP, Apache Tomcat เพื่อสร้างบริการ LDAP และ Web Service ให้กับ Application ใดๆ ที่ต้องการใช้งาน SPA-MFA สามารถเรียกใช้งานบริการของ LDAP และ Web Service เพื่อยืนยันตัวตนผู้ใช้ได้ทันที ผ่าน LDAP และ SOAP/XML และผู้ใช้สามารถ Download โปรแกรม J2ME ติดตั้งลงที่โทรศัพท์มือถือก็จะสามารถทำ TFA ได้ง่าย ซึ่งแตกต่างจากข้อเสนอ [2], [3], [5],[12], [21],[20] ที่ต้องทำการติดตั้ง Server เฉพาะหรืออุปกรณ์เฉพาะเพื่อใช้งาน เช่น AuthAnvil Server , FireID MAS Server, Diversinet SMS Server, Smard Reader และ RAS ACE/Server เป็นต้น จะเห็น

ได้ว่าการออกแบบระบบ SPA-MFA ในส่วนนี้ สามารถแก้ปัญหาเรื่อง ความยุ่งยากและข้อจำกัดในการติดตั้งระบบเพื่อใช้งานของข้อเสนอต่างๆ ก่อนหน้านี้ได้

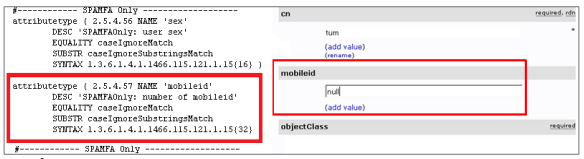
3.4 ในแง่ราคา (Cost)

ได้มีการเลือกใช้เครื่องมือในการพัฒนาที่เป็น Open Source Software ทั้งหมด คือ ใช้ภาษา Java และ ใช้โทรศัพท์มือถือที่ผู้ใช้พกพาอยู่แล้ว ทำให้การนำระบบไปใช้งานไม่ต้องมีค่าใช้จ่ายในเรื่องของ Security Token ดังที่ได้กล่าวไปในข้อที่ และ ค่าใช้จ่ายของ Server เฉพาะ ดังที่กล่าวไปในข้อที่ 3.1.3

3.5 ใช้อุปกรณ์ที่เหมาะสม

ออกแบบให้ใช้โทรศัพท์มือถือเป็นอุปกรณ์สำหรับ User Possession ซึ่งข้อมูลจาก [11] กล่าวไว้ว่าในเดือนกุมภาพันธ์ พ.ศ. 2552 มีผู้ใช้งานโทรศัพท์มือถือทั่วโลกถึง 3,300 ล้านเครื่อง และข้อมูลจาก [13] กล่าวว่า มีผู้ใช้งานโทรศัพท์มือถือในประเทศไทยถึง 59 ล้านเครื่องจากประชากรทั้งหมด 65 ล้านคน จะเห็นได้ว่ามีการใช้งานโทรศัพท์มือถืออย่างแพร่หลาย ทั้งนี้โทรศัพท์มือถือดังกล่าวส่วนใหญ่ต้องสนับสนุนการทำงานของ J2ME ได้ ซึ่งส่วนใหญ่สนับสนุนเทคโนโลยีดังกล่าว

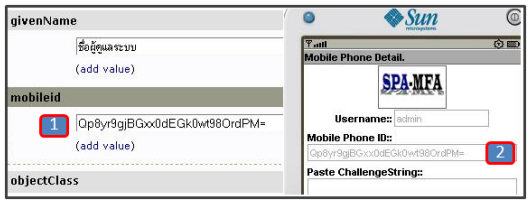
4. ผลการพัฒนาโปรแกรมต้นแบบ



รูปที่ 2 ผลการพัฒนา LDAP Server

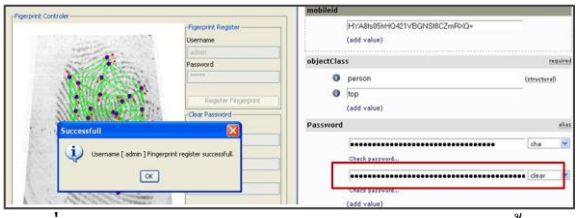
จาก รูปที่ 2 แสดงผลการเพิ่ม Schema ของ SPA-MFA ลงบน Schema ของ LDAP ในส่วนของ ObjectClass person โดยอ้างอิงจากมาตรฐาน RFC 2556 [22] ผลการทำงาน คือทำให้ SAP-MFA มีการจัดเก็บข้อมูลแบบ LDAP และเพิ่มเติมส่วน ObjectClass person ให้มี Attributer เพิ่มขึ้นก็คือ mobileid ซึ่งใช้เก็บข้อมูล MobileID ในขั้นตอนการลงทะเบียนโทรศัพท์มือถือที่จะนำมาใช้เป็น User Possession ใน SPA-MFA โดยการพัฒนา LDAP Server เลือกใช้ OpenLDAP และ phpLDAPadmin version 1.2.1.1 สำหรับจัดการข้อมูลใน LDAP และได้นำเทคโนโลยี SSL เพื่อเข้ารหัสข้อมูลที่ส่งผ่าน LDAP ที่ Port หมายเลข 636 ซึ่งได้ผลการพัฒนาดังรูปที่ 3

รูปที่ 3 ผลการพัฒนา LDAP และ phpLDAPadmin



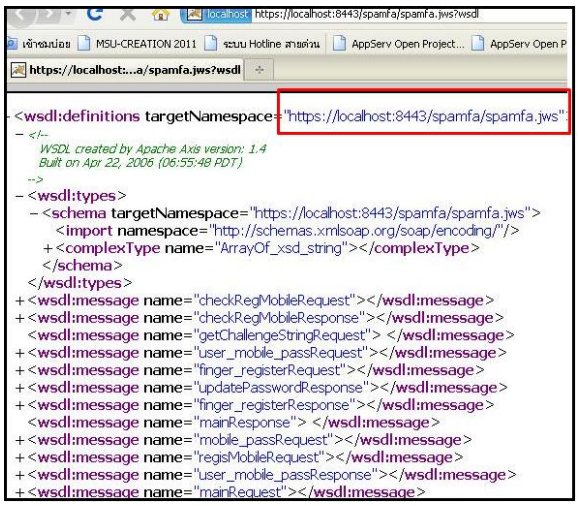
รูปที่ 4 ผลการพัฒนาโปรแกรมส่วนการลงทะเบียนโทรศัพท์มือถือ

จากรูปที่ 4 แสดงผลการพัฒนาในส่วนลงทะเบียนโทรศัพท์มือและซอฟต์แวร์ J2ME ด้วยการกำหนด MobiileID ให้กับโทรศัพท์มือถือของผู้ใช้ซึ่ง MobileID จะได้รับจาก Server เมื่อผู้ใช้ส่งข้อมูลการลงทะเบียนมือถือให้ Server คือ Username/Password และหมายเลขโทรศัพท์ หาก Username/Password ถูกต้อง Server จึงจะแสดง MobileID ซึ่งจากรูปที่ 4 *หมายเลข 1* แสดงค่า MobileID ที่เก็บไว้ที่ LDAP และ *หมายเลข 2* คือ ค่าที่เก็บในโทรศัพท์มือถือของผู้ใช้ และเมื่อลงทะเบียนเสร็จสิ้น ซอฟต์แวร์ J2ME จึงจะสามารถใช้ฟังก์ชั่นอื่นต่อไปได้ ในส่วนนี้ทำให้ระบบ SPA-MFA มี MobileID และหมายเลขโทรศัพท์มือถือของผู้ใช้ โดยจะถูกจับคู่กับ ชื่อผู้ใช้ หรือ cn แบบ 1 ต่อ 1 และ MobileID จะนำไปเป็น Seed ร่วมกับ Challenge จาก Server เพื่อสร้าง OTP ต่อไป จะทำให้ OTP ที่ได้แตกต่างกันไป ตาม MobileID



รูปที่ 5 ผลการพัฒนาระบบในส่วนการลงทะเบียนลายนิ้วมือ

จากรูปที่ 5 แสดงผลการลงทะเบียนลายนิ้วมือและจัดเก็บไว้ที่ LDAP โดยใช้ Java และเครื่องอ่านลายนิ้วมือ รุ่น U.are.U 4000 Sensor ผลการทำงานของโปรแกรมคือ ผู้ใช้สแกนลายนิ้วมือที่เครื่อง โปรแกรมจะแสดงเมนูในการลงทะเบียนลายนิ้วมือ โดยผู้ใช้จะต้องใส่ ชื่อผู้ใช้และรหัสผ่านที่ได้ลงทะเบียนไว้ หากผู้ใช้และรหัสผ่านถูกต้อง โปรแกรมจะส่งลายนิ้วมือนั้นไปเก็บไว้ที่ LDAP เพื่อใช้ในกระบวนกู้คืนรหัสผ่านในกรณีที่ผู้ใช้ลืมรหัสผ่านต่อไป



รูปที่ 6 ผลการพัฒนา Web Service ของ SPA-MFA

รูปที่ 6 แสดงไฟล์ WSDL ซึ่งเป็นผลการพัฒนา Web Service ในระบบ SPA-MFA เพื่อให้บริการ กับ Application ต่างๆ ที่ต้องการใช้งานการยืนยันตัวตนแบบพหุปัจจัย ใน SPA-MFA ซึ่งเป็นอีกช่องทางหนึ่ง ในการใช้บริการ SPA-MFA นอกเหนือจากช่องทางของ LDAP ทำให้ Application ต่างๆ นั้นสามารถเลือกช่องทางใช้งานได้หลากหลายขึ้น พัฒนาโดยใช้ Apache Tomcat 5.5.20 และ Apache Axis 1.4 เป็นเครื่องมือในการสร้างและให้บริการยืนยันตัวตนผู้ใช้ทั้งการยืนยันตัวตนเข้าใช้ระบบ และการยืนยันตนเพื่อกู้คืนรหัสผ่านและได้นำเทคโนโลยี SSL เพื่อเข้ารหัสข้อมูลที่ส่งผ่าน SOAP/XML ที่ Port หมายเลข 8443 ร่วมกับการพัฒนาในครั้งนี้ ทำให้ข้อมูลที่ส่งผ่านกันระหว่าง Application กับ Server มีการเข้ารหัสข้อมูล

รูปที่ 7 ผลการพัฒนาส่วน User Login

จากรูปที่ 7 แสดงการยืนยันตัวส่วน User Login โดย Server กำหนด Challenge String ให้กับผู้ใช้เพื่อนำ Challenge String พิมพ์ลงในโปรแกรมบนโทรศัพท์มือถือสำหรับการสร้าง MPP และนำ MPP ที่ได้ส่งกลับให้ Server มีการ Login ได้ 3 แบบ คือ (1) ใช้ข้อมูลสองส่วนร่วมกันคือ รหัสผ่าน มาจาก User Knowledge และ MPP มาจาก User Possession คือ โทรศัพท์มือถือที่ตั้งโปรแกรม J2ME และได้ลงทะเบียนไว้แล้ว ซึ่งการใช้รูปแบบข้อมูลสองส่วนร่วมกันนี้จะถือว่าปลอดภัยกว่ารูปแบบอื่น (2) ใช้ชื่อผู้ใช้รวมกับรหัสผ่าน คือ ยังคงการ Login แบบเดิมไว้ในกรณีที่โทรศัพท์มือถือหายหรือรำรุด ซึ่งจะมีประโยชน์ในการสลับใช้งานหากเกิดปัญหาขึ้นที่โทรศัพท์มือถือทำให้สามารถแก้ปัญหาเรื่อง การสูญหาย อายุการใช้งาน ที่เป็นข้อเสียของการ User Possession ได้ (3) ใช้ MPP คือ ใช้เพียง OTP ที่แสดงบนโทรศัพท์มือถือเท่านั้น เป็นทางเลือกหนึ่งที่ผู้ใช้สามารถ Login ได้ด้วย MPP จะมีข้อดีคือ รหัสผ่านไม่รั่วไหล ข้อเสียคือ หากมือถือตกไปอยู่ในมือผู้อื่น รหัสผ่านที่แสดงจะนำไปใช้ Login ได้ นอกจากนั้นการสร้างและตรวจสอบ OTP สามารถแสดงผล Output ได้ 2 แบบ คือ S/Key และ BASE 64 [21] ซึ่งลักษณะ Output ของ BASE 64 จะใช้ผลจาก Output ของ FoldTo64Bit Function ซึ่งเป็น Function ของ S/Key (อธิบายไว้ใน Appendix A ของ RFC 2289) มาแปลงเป็น Output ด้วย BASE64 Data Encoding [21] ได้การแสดงผลเป็น 11 ตัวอักษร จากตัวอักษรและตัวเลขที่เป็นไปได้ทั้งหมด 64 ตัว (A-Z, a-z, 0-9, +, /) จากการ

พัฒนา Output แบบ BASE64 ได้ทำการวิเคราะห์ความเสี่ยงจาก Brute Force Attack เมื่อเทียบกับ OTP แบบต่างๆ ดังนี้

ตาราง 1 ความเสี่ยงจาก Brute Force Attack ของ OTP แต่ละแบบ

OTP Type	Choice	Output	Probability (P)	Log10(P)	
HOTP/TOTP	10	6	10,000,000	7	10^7
HOTP/TOTP	10	7	100,000,000	8	10^8
HOTP/TOTP	10	8	1,000,000,000	9	10^9
HOTP/TOTP	10	19	10,000,000,000,000,000,000	19	10^19
S/Key	2048	6 [4-24]	73,786,976,294,838,206,464	19.86798	10^19
BASE64	64	11	73,786,976,294,838,206,464	19.86798	10^19

จากตาราง 1 แสดงความเสี่ยง Brute Force Attack ซึ่งดูจากความเป็นได้ของ Output ทั้งหมดที่จะเกิดขึ้น เห็นได้ว่าความเป็นได้ที่จะเกิดผลทั้งหมดของ BASE64 = 64^{11} (มาจากความเป็นไปได้ (P) ที่จะเกิด BASE64 OTP หนึ่งหลักคือ 64 : ตามตารางที่ 1 ใช้ทั้งหมด 11 หลัก) หรือประมาณ 10^{19} จากข้อมูลข้างต้นจะเห็นได้ว่า P ของ BASE64 และ S/Key เท่ากัน แต่ BASE64 แสดงผลตัวอักษรในจำนวนที่น้อยกว่า คือ เพียง 11 ตัวอักษรเท่านั้น และนอกจากนี้ค่า P ของ BASE64 ยังมากกว่า HOTP และ TOTP ซึ่งหากจะทำให้ค่า P ของ HOTP และ TOTP ใกล้เคียงกับ BASE64 ต้องเพิ่มจำนวนตัวอักษร Output ของ OTP ทั้งสองแบบไปถึง 19 ตัวอักษรดังตารางที่ 1 และจากความเป็นไปได้ที่จะเกิด OTP ทั้งหมดที่มีมาก ทำให้การ Brute-force Attack ทำได้ยากขึ้น ด้วยเหตุเองทำให้ BASE64 มีความสามารถการป้องกันการ Brute-force Attack ได้เทียบเท่ากับ S/Key แต่มีจำนวน Output ของ OTP น้อยกว่า และยังปลอดภัยกว่า HOTP และ TOTP

รูปที่ 8 ผลการพัฒนาการกู้คืนรหัสผ่านด้วยลายนิ้วมือ

จากรูปที่ 8 เป็นการทำงานของ Finger Printer Software ซึ่งผู้ใช้สามารถกู้คืนรหัสผ่านได้ด้วยการสแกนลายนิ้วมือที่เครื่องอ่านลายนิ้วมือพร้อมกับตั้งรหัสผ่านใหม่ที่ต้องการและเลือก Change Password ที่โปรแกรม ที่ตัวโปรแกรมจะส่งข้อมูลลายนิ้วมือไปที่ LDAP Server เพื่อนำมาตรวจสอบข้อมูลลายนิ้วมือที่ได้ลงทะเบียนไว้ หากข้อมูลถูกต้องระบบจะทำการตั้งรหัสผ่านใหม่ที่ผู้ใช้ต้องการให้

และขั้นตอนการกู้คืนรหัสผ่านด้วย SMS ทำได้โดยผู้ใช้ร้องส่ง SMS ตามรูปแบบที่ระบบกำหนดดังนี้ *#User#NewPass#* โดย User คือ ชื่อผู้ใช้ และ NewPass คือ รหัสผ่านใหม่ที่ต้องการ เมื่อระบบตรวจสอบพบข้อความใหม่ใน GSM Modem จะนำข้อมูล Username จาก SMS และหมายเลขโทรศัพท์ที่ส่งข้อความมา ไปตรวจสอบกับข้อมูลในฐานข้อมูลหากข้อมูลตรงกันแสดงว่าชื่อผู้ใช้ดังกล่าวใช้โทรศัพท์มือถือที่ได้ลงทะเบียนไว้ ส่งมาร้องของกู้คืนรหัสผ่านระบบก็จะทำการเปลี่ยนรหัสผ่านนั้นให้กับผู้ใช้

5. วิเคราะห์ผลการพัฒนา

(1) การแก้ปัญหาการยืนยันตัวตนหลายจุด SPA-MFA สามารถยืนยันตัวตนผู้ใช้งานระบบต่างๆ ที่แตกต่างกันได้ด้วยรหัสผ่านเพียงตัวเดียว ผู้ใช้งานจึงไม่ต้องรับผิดชอบรหัสผ่านหลายตัวหากต้องการใช้งานหลายๆ ระบบ จึงสามารถแก้ไขปัญหาจากข้อเสนอก่อนหน้านี้คือ FiveBarGate [4], VeriSign Identity Protection (VIP) [6], Cellular Authentication Token [7] , Mobile One Time Passwords [8], 1Key [9]

(2) การแก้ปัญหาการรั่วไหลของรหัสผ่าน ระบบ SPA-MFA ใช้งานรหัสผ่านถึงสองส่วนในการยืนยันตัวตนคือรหัสผ่านจาก User Knowledge และ MPP ที่เป็น OTP จาก User Possession (โทรศัพท์มือถือ) ทำให้รหัสผ่านที่ใช้ยืนยันตัวตนนั้นมีการเปลี่ยนแปลงทุกครั้ง นั้นคือหากมีการรั่วไหลหรือถูกดักจับรหัสผ่านส่วนใดส่วนหนึ่งไป รหัสผ่านดังกล่าวจะไม่สามารถนำไปใช้ ยืนยันตัวตนเข้าสู่ระบบซ้ำได้อีก

(3) การแก้ปัญหาการลืมรหัสผ่าน ระบบ SPA-MFA ได้เสนอกระบวนการกู้คืนรหัสผ่านใหม่ โดยใช้ User Attribute คือ ลายนิ้วมือ ซึ่งเป็นข้อมูลที่ไม่ลืม ไม่รั่วไหล ทำให้สามารถแก้ปัญหาที่เกิดขึ้นจากการใช้ คำถามกันลืม และ E-mail Recover Password อีกทั้งกระบวนการดังกล่าวเป็นหน้าที่ของผู้ใช้เอง จึงไม่เกิดภาระกับผู้ดูแลระบบ และนอกจากนั้นได้มีการเพิ่มช่องทางในการกู้คืนรหัสผ่านโดยใช้ SMS เพื่อเป็นอีกช่องทางหนึ่งให้กับผู้ใช้ในกรณีที่ยังขาดเครื่องอ่านลายนิ้วมือ

(4) การแก้ปัญหาของข้อเสนอก่อนหน้านี้

- ปัญหาความไม่เหมาะสมกับการเลือกใช้อุปกรณ์ ในข้อเสนอ Smart Card Authentication [20] และ RSA SecuID [12] แก้ไขได้โดยการนำ โทรศัพท์มือถือที่ผู้ใช้พกพาอยู่แล้วมาใช้งานเป็น Security Token แทน

- ปัญหาการยืนยันอุปกรณ์ที่ใช้ ในข้อเสนอ FiveBarGate [4], HOTP [17], 1Key [9], Cellular Authentication Token [7], AuthAnvil [2], Mobile One Time Passwords [8], RSA SecurID [12] แก้ไขได้โดยการกำหนดให้มีการลงทะเบียนโทรศัพท์มือถือ ด้วย Mobile ID ที่ให้คู่กับ Username และโทรศัพท์มือถือแบบ 1 ต่อ 1 เพื่อให้ Mobile ID ดังกล่าวเป็นข้อมูลในการยืนยันอุปกรณ์

- ปัญหาความเป็นอิสระต่อระบบปฏิบัติการ ในข้อเสนอ AuthAnvil [2], FiveBarGate [4], Diversinet [5], Cellular Authentication Token [7] แก้ไขได้โดยการออกแบบระบบที่เป็น OpenLDAP, Java Web Service, Java Software และ J2ME ทำให้สามารติดตั้งระบบได้กับทุกๆ ระบบปฏิบัติการ

- ความยุ่งยากและข้อจำกัดในการติดตั้งระบบ ในข้อเสนอ Smart Card Authentication [20], RSA SecurID [12], AuthAnvil [2], Diversinet [5] แก้ไขโดย SPA-MFA ที่ต้องการติดตั้ง Web Service Server (Apache tomcat + Axis), LDAP Server (OpenLDAP), phpLDAPadmin เท่านั้น โปรแกรมติดตั้งส่วนต่างๆ นี้สามารถ Download ได้ฟรี และติดตั้งง่ายไม่ต้องใช้อุปกรณ์หรือ Server เฉพาะเพิ่มเติม

-ปัญหาจำนวนอักษรของ OTP ไม่เหมาะสม ในข้อเสนอ HOTP, TOTP และ S/Key แก้ไขโดยการใช้ BASE64 ดังที่ดังแสดงในตาราง 1

- ปัญหาเรื่องค่าใช้จ่ายในการติดตั้งใช้งานระบบ ในข้อเสนอ Aradiom SolidPass [1], AuthAnvil [2], FireID [3], FiveBarGate [4], Smart Card Authentication [20], RSA SecurID [12], Diversinet [5] แก้ไขโดยการเลือกใช้ Open Source Software และใช้โทรศัพท์มือถือเป็น Security Token จึงไม่จำเป็นต้องเสียค่าลิขสิทธิ์ Software และ ค่าใช้จ่ายในการซื้อ Token เพิ่มเติม

6. สรุปผลการพัฒนา

งานวิจัยนี้แสดงให้เห็นว่า การนำกระบวนการยืนยันตัวของผู้ใช้ในระบบต่างๆ เข้ามาตรวจสอบ ณ จุดเดียวทำให้ช่วยลดภาระและลดจำนวน รหัสผ่านที่ผู้ใช้ต้องรับผิดชอบลงได้เหลือเพียงหนึ่งรหัสผ่านเท่านั้น และการนำเทคนิคการผสมผสานปัจจัยการยืนยันตัวตน ระหว่าง User Knowledge ซึ่งเป็นที่นิยมและใช้กันอย่างแพร่หลาย ให้มีการทำงานงานร่วมกับ User Possession และ User Attribute ซึ่งเป็นปัจจัยที่สองและสามโดย User Possession ที่เลือกใช้นั้นได้เน้นความสำคัญไปที่ต้นทุนและเป็นที่พกพาอยู่แล้วของผู้ใช้งานระบบ ซึ่งได้เสนอปัจจัยดังกล่าวอยู่ในรูปแบบโทรศัพท์มือถือ ซึ่งนับว่ามีต้นทุนต่ำและมีพกพาอยู่มากหากเทียบกับ User Possession อื่นๆ ที่มีอยู่ในปัจจุบัน ทำให้สามารถแก้ไขปัญหาการดักจับข้อมูลและการรั่วไหลของข้อมูลได้ เนื่องจากรหัสมีถึงสองส่วนด้วยกันและถ้าขาดส่วนใดส่วนหนึ่งก็จะไม่สามารถใช้ในกระบวนการยืนยันตัวตนได้ และ User Attribute ที่เลือกคือลายนิ้วมือ ซึ่งผู้ใช้ไม่ลืมและไม่รั่วไหล เพื่อใช้ในกระบวนการกู้คืนรหัสผ่านพร้อมกับใช้ SMS ซึ่งเป็นอีกทางเลือกหนึ่งในการกู้คืนรหัสผ่านผ่านทาง User Possession ทำให้กระบวนการกู้คืนรหัสผ่าน มีความมั่นคงและน่าเชื่อถือมากขึ้น รวมไปถึงการออกแบบ SPA-MFA ที่สามารถนำไปประยุกต์ใช้งานได้ง่าย และสามารถแก้ไขจุดอ่อนในข้อเสนอที่มีมาก่อนหน้างานวิจัยฉบับนี้ได้เป็นอย่างดี

7. ข้อเสนอแนะงานวิจัย

พัฒนาระบบการยืนยันตัวตนใหม่ที่อาศัยการใช้ User Attribute ลักษณะอื่นๆ ที่เหมาะสม เช่น รูปแบบของใบหน้า ใช้รูปแบบเสียง เป็นต้น เนื่องจากในปัจจุบัน พ.ศ. 2555 มีการพัฒนา Algorithm และวิธีการในการยืนยันตัวตนด้วยรูปแบบของใบหน้า Face Authentication ซึ่งที่โดดเด่นที่สุดในตอนนี้ คือ การยืนยันตัวเข้าใช้คอมพิวเตอร์โน้ตบุ๊กด้วยใบหน้าที่ถ่ายจาก

กล้องเว็บแคมบนเครื่อง ซึ่งไม่จำเป็นต้องใช้อุปกรณ์เพิ่มเติม ดังนั้นควรที่จะมีการศึกษา และปรับปรุงความสามารถในการตรวจสอบความเป็นจริงของใบหน้าผู้ใช้ เพื่อนำ User Attribute นี้มาประยุกต์กับระบบสารสนเทศในปัจจุบัน จะสามารถทำให้กระบวนการยืนยันตัวตนมั่นคงมากขึ้น

8. กิตติกรรมประกาศ

งานวิจัยฉบับนี้ สำเร็จเรียบร้อยได้ด้วยความเสียสละและความอนุเคราะห์ทุนวิจัยจากสำนักศึกษาทั่วไป มหาวิทยาลัยมหาสารคาม ที่สนับสนุนการศึกษาและทำวิจัย จึงขอกราบขอบพระคุณในความกรุณามา ณ โอกาสนี้

เอกสารอ้างอิง

[1] "Aradiom SolidPass", http://www.aradiom.com/SolidPass.
[2] "AuthAnvil", http://www.scorpionsoft.com/.
[3] "FileID", http://www.fireid.com/.
[4] "FiveBarGate", http://www.fivebargate.net/.
[5] "Diversinet", http://www.diversinet.com.
[6] "VeriSign Identity Protection", http://www.symantec.com/business/.
[7] "Cellular Authentication Token", http://www.megaas.com/.
[8] "Mobile One Time Passwords", http://motp.sourceforge.net
[9] "1Key", http://www.rho.cc/index.php/software/1key.
[10] "Supported fingerprint scanners and sensors", http:// www.neurotechnology.com/cgibin/fingerprint-scanners.cgi.
[11] "Mobile Popular", http://www.reuters.com/.
[12] "Strong Two Factor Authentication with RSA SecurID", SSH Communications Security Corp, Design Solution Report, November 2008.
[13] "Communication technology trend and market report Quarter 3", The National Telecommunication Commission(NTC), Market Report 3, May 2008.
[14] M. Benantar, Access control systems: security identity management and trust models, 2nd ed. Texas: Springer, 2006.
[15] M. Bishop, Computer Security Art and Science, 1st ed. Massachusetts: Addison Wesley Professional, 2003.
[16] M. Curtin, Brute force: cracking the data encryption standard, 1st ed. New York: Springer, 2005.
[17] D. M'Raihi, M. Bellare, F. Hoornaert, D. Naccache, and O. Ranen, "HOTP: An HMAC-Based One-Time Password Algorithm", IETF, RFC 4226, December 2005.
[18] S. Garfinkel, G. Spafford, and A. Schwartz, Practical UNIX and Internet Security, 3rd ed. California: O'Reilly Media, Inc, 2003.
[19] K. Hausman and S. Cook, IT Architecture for Dummies, 3rd ed. New Jersey: Wiley Publishing Inc, 2011.
[20] M. Hendry, Smart Card Security and Applications, 2nd ed. Washington D.C: Artech House, 2001..
[21] S. Josefsson, "The Base16 Base32 and Base64 Data Encodings", IETF, RFC 4648, October 2006.
[22] M.Wahl, "A Summary of the X.500(96) User Schema for use with LDAPv3", IETF, RFC 2256, December 1997.
[23] N. Haller, C. Metz, P. Nesser, and M. Straw, "A One-Time Password System", IETF, RFC 2289, February 1998.
[24] T. Plum and R. Bleiler, User authentication, 1st ed. Washington D.C: SPEC Kits, 2001.
[25] D. M. Raihi, S.Machani, M.Pei, and J.Rydell, "TOTP: Time-Based One-Time Password Algorithm", IETF, RFC 6238, May 2011.
[26] J. Sermersheim, "Lightweight Directory Access Protocol (LDAP): The Protocol", IETF, RFC 4511, June 2006.