**แนวทางการป้องกันการโจมตีข้อมูลจากเอสคิวแอลอินเจคชันด้วยการกำหนดสิทธิ์การเข้าถึงข้อมูล**

**An Approach to Prevent SQL Injection Attacks Using Granting Permissions**

*ธนากร จงใจงาม (thanakorn jongjaingam)1 กรวีร์ เหล่าศรี (korawee laosri)2*

*ฉัตรบุศ ตระกูลนิ่มนวล (chattabut trakulnimnual)3และชุมพล โมฆรัตน์ (Chumpol Mokarat)4*

*สาขาวิชาเทคโนโลยีสารสนเทศ คณะบริหารธุรกิจและเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีราชมงคลตะวันออก*

*1thanakorn.jon@rmutto.ac.th 2korawee.lao@rmutto.ac.th 3chattabut.tra@rmutto.ac.th 4chumpol\_mo@rmutto.ac.th*

**คำสำคัญ:** แอปพลิเคชันการเช่ารถ, กำหนดสิทธิ์การเข้าถึงข้อมูล, ความมั่นคงปลอดภัยของข้อมูล, เอสคิวแอลอินเจคชัน

**วัตถุประสงค์การดำเนินงาน :**

1.เพื่อควบคุมการเข้าถึงของผู้ใช้งาน หรือกำหนดสิทธิ์การใช้งานเพื่อเข้าถึงข้อมูล

2.เพื่อรักษาความปลอดภัยของฐานข้อมูลและป้องกันข้อมูลที่เป็นความลับ

3.เพื่อลดความเสี่ยงความเสียหายของข้อมูลจากการโดน Hack

4.เพื่อศึกษาการโจมตีข้อมูลและวิธีการป้องกันการโจมตีข้อมูลในรูปแบบ SQL Injection

**1.บทนำ**

การให้บริการเดิมของเต๊นท์เช่ารถยนต์คือ เมื่อลูกค้าจะทำการเช่ารถยนต์ในแต่ละครั้ง จะต้องมาดูรถที่เต๊นท์รถด้วยตัวเองซึ่งอาจมีระยะทางและค่าเดินทางค่อนข้างมากหากอยู่ไกล และเมื่อต้องการเช่าต้องเตรียมเอกสารมากมายเช่น บัตรประชาชน ใบอนุญาตขับขี่ หรือเอกสารต่างๆมาให้ทางร้าน ซึ่งบางครั้งหากเตรียมเอกสารไม่ครบจะทำให้เสียเวลาในการกลับไปเอาเอกสาร การเก็บเอกสารจำนวนมากมีความเสี่ยงที่เอกสารสำคัญจะสูญหายหรือยากต่อการค้นหาข้อมูลในแต่ละครั้ง และการเช่านั้นจะต้องมีพนักงานในการบริการ ซึ่งการบริการแต่ละครั้งก็จะต้องใช้พนักงานหนึ่งต่อลูกค้าหนึ่งคน ทำให้มีพนักงานไม่เพียงพอทำให้เกิดปัญหาความล่าช้าในการให้บริการกับลูกค้าเช่ารถยนต์ พนักงาน 1 คนได้ทำการเช่ารถยนต์หลายครั้ง จึงทำให้เกิดการสับสนหรือปัญหาในเรื่องของการให้เช่ารถผิดคัน จะเห็นได้ว่าการทำงานในระบบเดิมนั้นจะใช้แรงงานคนในทุกขั้นตอนการทำงาน

รายงานการศึกษานี้จึงได้พัฒนาระบบเพื่อนำไปช่วยสนับสนุนการเช่ารถยนต์ของเต๊นท์เช่ารถให้มีประสิทธิภาพมากยิ่งขึ้น ข้อมูลทั้งหมดจะถูดจัดเก็บไว้ภายในฐานข้อมูล เพื่อสะดวกต่อการนำข้อมูลมาใช้งาน เพื่อป้องกันการเข้าถึงข้อมูลที่เป็นข้อมูลลับจึงต้องกำหนดสิทธิ์การเข้าถึงข้อมูลให้ผู้ใช้งานในตารางผู้ใช้งานโดยแบ่งออกเป็น3กลุ่มผู้ใช้งาน ได้แก่ กลุ่มแอดมิน กลุ่มผู้เช่า และกลุ่มผู้ให้เช่า

การพัฒนาระบบงานเว็บแอพพลิเคชั่น นักพัฒนาส่วนใหญ่จะมุ่งเน้นที่การพัฒนาระบบงานโดยอาจละเลยหรือไม่ได้ให้ความสนใจในเรื่องความปลอดภัยเท่าที่ควร ทำให้ผู้ไม่ประสงค์ดีสามารถโจมตีเว็บแอพพลิเคชั่น ในขั้นตอนการตรวจสอบการเข้าสู่ระบบ (Authentication) หรือป้อนคำสั่งโจมตีผ่านทางฟอร์มกรอกข้อมูลต่าง ๆ การโจมตี SQL Injection เกิดจากช่องรับข้อมูลต่างๆของเว็บเพจไม่สมบูรณ์จึงเป็นจุดอ่อนหรือช่องโหว่ที่Hackerสามารถส่ง SQL Injection เข้ามาในช่องรับข้อมูลเพื่อโจมตีข้อมูล จึงทำการศึกษาวิธีการโจมตีของ และวิธีการป้องกันข้อมูล ข้อดีของการป้องกันคือป้องกันการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต ทำให้ข้อมูลไม่สูญหายและระบบยังสามารถทำงานได้อย่างต่อเนื่อง ข้อเสียคือไม่สามารถป้องกันการโดนHackข้อมูลได้100เปอร์เซ็น อาจต้องมีการตรวจเช็คระบบและอัพเดตให้เป็นแพตช์ล่าสุดอยู่เสมอ

ดังนั้น เพื่อเป็นการแก้ไขปัญหาดังกล่าวผู้ศึกษาจึงทำระบบป้องกันการโจมตีข้อมูลจากเอสคิวแอลอินเจคชันด้วยการกำหนดสิทธิ์การเข้าถึงข้อมูลเพื่อแก้ไขปัญหาการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาตด้วยเครื่องมือ PDO ฟังก์ชันของ PHP โดยเลี่ยงการเอาตัวแปรที่รับค่าจากผู้ใช้งานจริง นำไปใส่คิวรี่ แก้โดยการนำตัวแปรที่รอรับข้อมูลจริงไปใส่ในคิวรี่แทน

**2.ทฤษฎีที่เกี่ยวข้อง**

**2.1 การกำหนดสิทธิ์ [1]**

2.1.1 การกำหนดสิทธิ์การเข้าถึง หมายถึง การควบคุมการเข้าถึงของผู้ใช้งานข้อมูลเป็นส่วนที่เจ้าของข้อมูลหรือหน่วยงานเพื่อควบคุมการเข้าถึงของผู้ใช้งาน หรือปิดสิทธิ์ไม่ให้ผู้ใช้งานเห็นข้อมูลนั้นๆ ผ่านการเข้ารหัสผ่าน การกำหนดขึ้นอยู่กับบทบาทของแต่ละบุคคล

2.1.2 access rights (แอคเซ็ซ ไรท์)  หรือสิทธิการเข้าถึง เป็นการกำหนดว่าให้บุคคลนี้เข้าถึงระบบได้แค่ไหน เห็นข้อมูลได้แค่ไหน สามารถกำหนดได้ว่าต้องการให้ลูกค้าคนนี้สามารถเข้ามาในระบบได้แค่ไหน ดูข้อมูลอะไรได้บ้าง และมีสิทธิทำอะไรได้บ้าง

**2.2 ความมั่นคงปลอดภัยของข้อมูล [2]**

**ความมั่นคงปลอดภัย (Security)** คือ สถานะที่มีความปลอดภัย ไร้กังวล อยู่ในสถานะที่ไม่มีอันตรายและได้รับการป้องกันจากภัยอันตรายทั้งที่เกิดขึ้นโดยตั้งใจหรือบังเอิญ เช่น ความมั่นคงปลอดภัยของประเทศ ย่อมเกิดขึ้นโดยมีระบบป้องกันหลายระดับ เพื่อปกป้องผู้น าประเทศ ทรัพย์สิน ทรัพยากร และประชาชนของประเทศ เป็นต้น **การรักษาความมั่นคงปลอดภัยของสารสนเทศ (Information Security)** คือผลที่เกิดขึ้นจาการใช้ระบบของนโยบายสำหรับการรักษาความมั่นคงปลอดภัยสารสนเทศ ควบคุม และป้องกันการเปิดเผย โดยไม่ได้รับอนุญาต **ภัยคุกคาม (Threat)** คือ วัตถุ สิ่งของ ตัวบุคคล หรือสิ่งอื่นใดที่เป็นตัวแทนของการกระทำอันตรายต่อทรัพย์สินขององค์กร หรือสิ่งที่อาจจะก่อให้เกิดเสียหายต่อคุณสมบัติของข้อมูลด้านใดด้านหนึ่งหรือมากกว่าหนึ่งด้าน ความลับ (Confidentiality), ความสมบูรณ์ (Integrity), ความพร้อมใช้ (Availability)

**2.3 องค์ประกอบของความมั่นคงปลอดภัยของข้อมูล [3]**

*ความลับ (Confidentiality)* เป็นการรับประกันว่าผู้มีสิทธิ์และได้รับอนุญาตเท่านั้นที่สามารถเข้าถึงข้อมูลได้ องค์กรต้องมีมาตรการป้องกันการเข้าถึงสารสนเทศที่เป็นความลับ *ความสมบูรณ์( Integrity)*  ความสมบูรณ์ ครบถ้วน ถูกต้อง เป็นสารสนเทศที่นำไปใช้ประโยชน์ได้อย่างถูกต้องครบถ้วน  *ความพร้อมใช้(Availability)*  สารสนเทศจะถูกเข้าถึงหรือเรียกใช้งานได้อย่างราบรื่น และใช้งานได้ต่อเนื่องตลอดเวลาไม่มีข้อบกพร่อง ความถูกต้องแม่นยำ (Accuracy) สารสนเทศต้องไม่มีความผิดพลาด และต้องมีค่าตรงกับความคาดหวังของผู้ใช้เสมอ *เป็นของแท้ Authenticity* สารสนเทศที่ถูกจัดทำขึ้นจากแหล่งที่ถูกต้อง ไม่ถูกทำซ้ำโดยแหล่งอื่นที่ไม่ได้รับอนุญาต หรือแหล่งที่ไม่คุ้นเคยและไม่เคยทราบมา *ความเป็นส่วนตัว* สารสนเทศที่ถูกรวบรวม เรียกใช้ และจัดเก็บโดยองค์กร จะต้องถูกใช้ในวัตถุประสงค์ที่ผู้เป็นเข้าของสารสนเทศรับทราบ ณ ขณะที่มีการรวบรวมสารสนเทศนั้น

**2.4 SQL Injection [4]**

SQL Injection (SQLi) เป็นการโจมตีแบบฉีดซึ่งผู้โจมตีดำเนินการคำสั่ง SQL ที่เป็นอันตรายเพื่อควบคุมเซิร์ฟเวอร์ฐานข้อมูลของเว็บแอปพลิเคชัน ดังนั้นจึงเข้าถึง แก้ไข และลบข้อมูลที่ไม่ได้รับอนุญาต รูปแบบการโจมตี SQL Injection ผู้โจมตีจะค้นหารูปแบบ Input ของผู้ใช้ที่มีช่องโหว่ภายใน Web Application ก่อน หลังจากนั้นผู้โจมตี จะส่ง Input ที่มีรูปแบบที่ถูกต้องตรงเข้าไปที่ Database โดยไม่ต้องผ่านหน้า Web application ปกติซึ่งถ้าทำได้ ผู้โจมตีจะสามารถส่ง คำสั่ง SQL ที่เป็นอันตรายจะถูกดำเนินการในฐานข้อมูล

**2.5 MySQL**

เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์โอเพนซอร์สบนพื้นฐานของ SQL ซึ่ง MySQL ได้รับการออกแบบและปรับให้เหมาะสมสำหรับเว็บแอปพลิเคชันและสามารถทำงานบนแพลตฟอร์มใดก็ได้ MySQL ทำงานเป็น database server และอนุญาตให้ผู้ใช้หลายคนจัดการและสร้างฐานข้อมูลจำนวนมาก มันเป็นองค์ประกอบสำคัญใน LAMP (LAMP ย่อมาจาก Linux, Apache, MySQL และ PHP)

Graphical user interface, text, application

Description automatically generated

**ภาพที่ 1** ภาพตัวอย่าง Mysql

**2.6 PHP Language**

PHP เป็นภาษาสคริปต์ ( Scripting Language ) คำสั่งต่างๆ จะเก็บในรูปแบบของข้อความ (Text)อาจเขียนแทรกอยู่ภายในภาษา HTML หรือใช้งานอิสระก็ได้ แต่ในการใช้งานจริงมักใช้งานร่วมกับภาษา HTML ดังนั้นการเขียนโปรแกรมนี้ต้องมีความรู้ด้านภาษา HTML เป็นอย่างดี อย่างไรก็ตามเราสามารถใช้โปรแกรมประยุกต์มาช่วยอำนวยความสะดวกในการสร้างงานได้

A screenshot of a computer

Description automatically generated with medium confidence

**ภาพที่ 2** ภาพตัวอย่างภาษา PHP

**3.งานวิจัยที่เกี่ยวข้อง**

**แววตา เตชาทวีวรรณ (2020) [5]** ศึกษาเกี่ยวกับความมั่นคงปลอดภัยของสารสนเทศ ตาม “แนวนโยบายและแนวปฏิบัติในการรักษาความมั่นคงปลอดภัยด้านสารสนเทศของหน่วยงานรัฐ โดยพบว่า ห้องสมุดสถาบันอุดมศึกษาของรัฐให้ความสำคัญต่อการรักษาความมั่นคงปลอดภัยระบบสารสนเทศโดยมีการปฎิบัติแนวทางในการรักษาความมั่นคงปลอดภัยด้านสารสนเทศ ด้วยวิธีทางอิเล็กทรอนิกส์ โดยกําหนดสิทธิและระดับของบุคคล ทั้งทางกายภาพและทางอิเล็กทรอนิกส์ การป้องกันการเข้าถึงอุปกรณ์ต่าง ๆ โดยกําหนดกฎระเบียบและใช้อุปกรณ์ป้องกันบุคคลอื่นเข้าถึงโดยไม่ได้รับอนุญาต ห้องสมุดจึงจําเป็นต้องมีมาตรการป้องกันรวมทั้งการควบคุมการเข้าถึงโปรแกรมประยุกต์หรือแอปพลิเคชันและสารสนเทศ โดยให้บุคลากรกําหนดชื่อผู้ใช้งาน (Username) และรหัสผ่าน (Password) สําหรับเครื่องคอมพิวเตอร์ที่ตนใช้ปฏิบัติงาน รวมทั้งติดตั้งโปรแกรม คอมพิวเตอร์ลิขสิทธิ์สําหรับใช้ในการปฏิบัติงานตามความจําเป็น

**กิตติศักดิ์ แก้วบุตรดี(2564)**[6] ได้ศึกษาเกี่ยวกับการดูแลข้อมูลให้มีความมั่นคงปลอดภัยและการดำเนินโครงการความมั่นคงปลอดภัยระบบสารสนเทศ (ISO/IEC27001: 2013) เพื่อกำกับดูแลระบบสารสนเทศ ให้มีความ มั่นคงและปลอดภัยตามมาตรฐานสากลประกอบพื้นฐานที่สำคัญ 3 ข้อ ความลับของข้อมูล (Confidentiality) ความพร้อมใช้งานของข้อมูลและระบบ สารสนเทศต่าง ๆ (Availability) ความถูกต้องสมบูรณ์ของข้อมูล (Integrity) โดยใช้หลักการวงจรบริหารคุณภาพ PDCA คือ Plan (การวางแผน) คือการวางแผน และ กำหนดขอบเขตการดำเนินการจัดตั้งโครงการ Do (การปฏิบัติ)คือการจัดทำเอกสารการ Check (การตรวจสอบ) คือการจัดทำเอกสารและตรวจสอบความถูกต้อง และ Act (การปรับปรุง) คือการติดตามข้อบกพร่อง ทบทวนและพิจารณาเพื่อใช้ในการวางแผนการดำเนินงานภายใต้กรอบของการในครั้งต่อไป

**ภุมวุฒิ วิทวัสสำราญกุล (2020)[7]** ศึกษาวิเคราะห์พบว่าแนวทางรักษาความปลอดภัยภายในศูนย์ปฏิบัติการเครือข่าย สามารถกำหนดแนวทางขั้นตอนปฏิบัติสำหรับอุปกรณ์ โดยการแบ่งประเภทอุปกรณ์และระบบที่เหมาะสมมีความสำคัญอย่างมากในการดำเนินการ ข้อดีพบว่าสามารถนำแนวทางรักษาความปลอดภัยดังกล่าวไปใช้จัดการความปลอดภัยข้อมูลได้อย่างรวดเร็ว และ ดำเนินการด้านการรักษาความปลอดภัยข้อมูล และ มีการดำเนินการตามมาตรฐาน ISO 27001 ข้อจำกัดของการศึกษาพบว่าไม่สามารถศึกษาซอฟต์แวร์ต่าง ๆ ที่ติดตั้งใช้งานภายในศูนย์ปฏิบัติการเครือข่ายฯ เนื่องจากหน่วยงานราชการมีการพัฒนาซอฟต์แวร์ ที่ไม่เป็นไปตามกระบวนการมาตรฐาน การศึกษาโดยใช้ข้อมูลของสินทรัพย์ข้อมูลสารสนเทศภายในศูนย์ปฏิบัติงานเครือบ่ายซึ่งมีความอ่อนไหว และวัฒนธรรมองค์กรเกี่ยวกับกฎหมายที่มีการกำหนดโดยพฤตินัยไว้ว่า หากยังไม่มีกฎระเบียบรองรับ หรือผ่านการพิจารณาของคณะกรรมการแล้วจะให้ยุติไว้ก่อน การขอข้อมูลเพื่อใช้ในการศึกษา และเผยแพร่จึงเป็นเรื่องยาก

**4. การดำเนินงานวิจัย**

ในการดำเนินวิจัยฉบับนี้เป็นการนำเสนอแนวทางการแก้ไขปัญหาการHack ข้อมูล 2 ที่คือ จาก Form การเข้าสู่ระบบเช่น การป้อนข้อมูลบางอย่างโดยวิธีการใส่คำสั่ง or’1’=’1 และการ Hack จาก URL โดยใส่คำสั่ง or’1=1’ ในช่อง URL จะนำเสนอแนวทางสำหรับการกำหนดสิทธิ์ผู้ใช้งานและการป้องกันข้อมูลจากการ Hack ด้วยการ PDO ของ PHP ซึ่งประกอบด้วยขั้นตอน ดังนี้ 1.ขั้นตอนการจัดเตรียมข้อมูลนำเข้าใน phpMyAdmin 2.เขียนโค้ด PHP เพื่อเชื่อมต่อกับข้อมูลใน phpMyAdmin และ 3.เขียนโค้ด PHP เพื่อป้องกันและทดสอบระบบการป้องกัน

ในการดำเนินงานการออกแบบและพัฒนาเว็บไซต์เช่ารถยนต์ ประกอบด้วยการดำเนินงานดังภาพ

Diagram

Description automatically generated

**ภาพที่ 3** แสดงภาพรวมการดำเนินงาน

การกำหนดสิทธิ์และการเข้าถึงข้อมูลมีเส้นทางการดำเนินงานดังรูป

Diagram

Description automatically generated

**ภาพที่ 4** ภาพ flow chart โดยรวมของระบบ

**1.แผนภาพอีอาร์ของระบบฐานข้อมูล**

Diagram

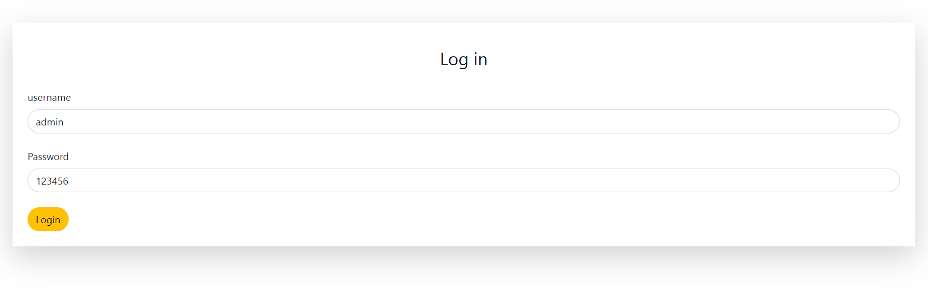
Description automatically generated

**ภาพที่ 5** แผนภาพอีอาร์ของระบบฐานข้อมูล

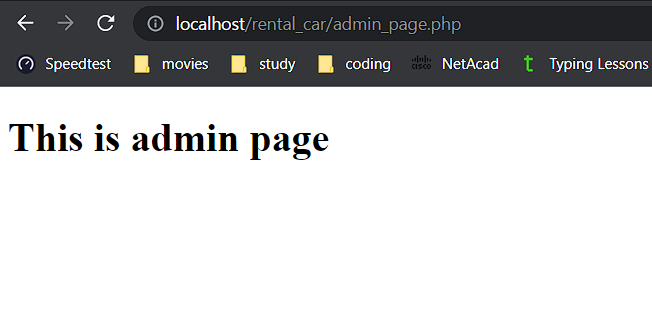
**3.เขียนโค้ดPHPเพื่อป้องกันและทดสอบระบบการป้องกัน**

ก่อนเขียนโค้ดPDO ฟังก์ชันของ PHP เพื่อป้องกันการ Hack ผ่าน Form การเข้าสู่ระบบ

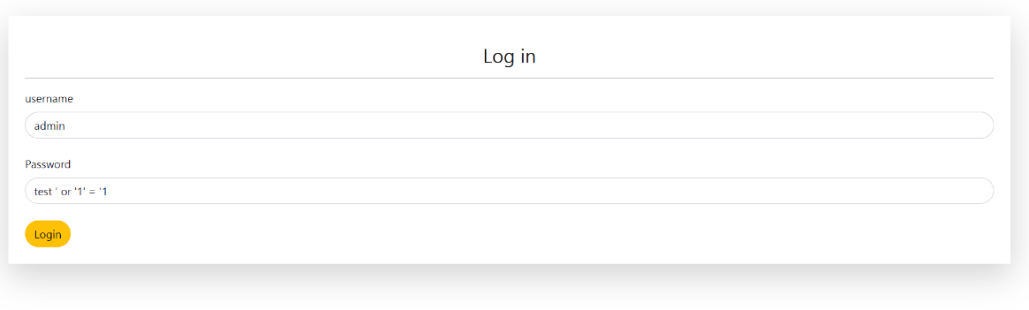
เข้าสู่ระบบปกติ ใส่username และ password ที่ถูกต้อง



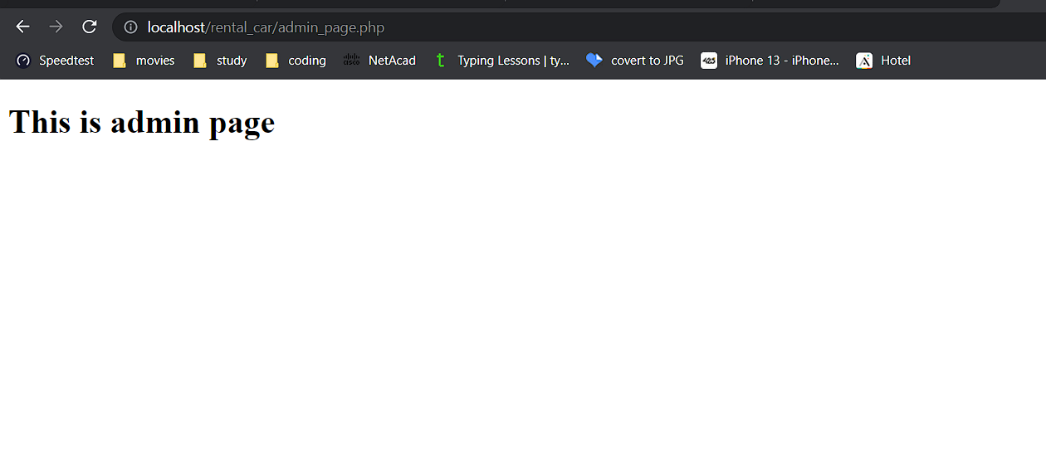
**ภาพที่ 6** แสดงหน้าเข้าสู่ระบบ



**ภาพที่ 7** แสดงหน้าแอดมินเพจ



**ภาพที่ 8** แสดงหน้าการเข้าสู่ระบบโดยใช้คำสั่ง or ‘1=1’

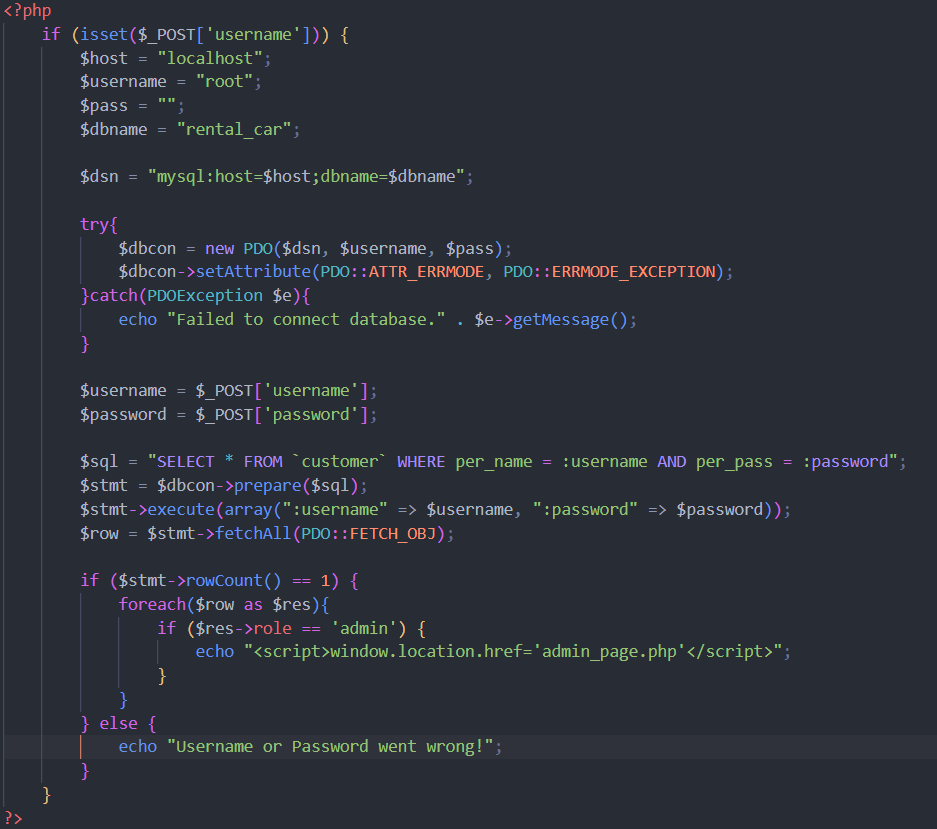


**ภาพที่ 9** แสดงหน้าแอดมินเพจหลังจากเข้าสู่ระบบโดยใช้คำสั่ง or ‘1=1’

จากภาพที่ 6 แสดงการเข้าสู่ระบบโดยการใส่ชื่อและรหัสผ่านที่ถูกต้องจะเข้าสู่ระบบหน้า admin ได้ตามปกติดังภาพที่ 7

จากภาพที่ 8 แสดงการเข้าสู่ระบบโดยการใส่คำสั่ง SQL Injection คือ or ‘1’= ‘1 จะสามารถเข้าสู่ระบบหน้า admin ได้ดังรูปที่ 9

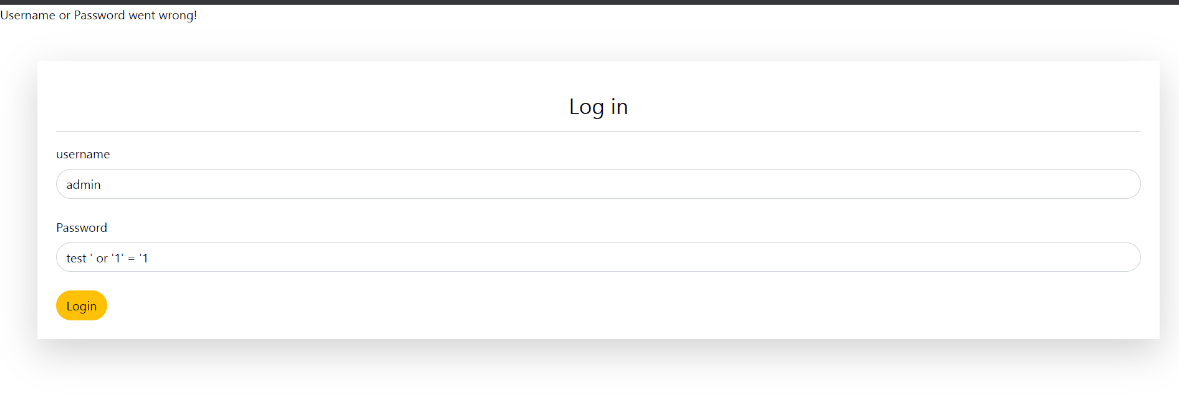
หลังจากเขียนโค้ด PDO ฟังก์ชันของ PHP เพื่อป้องกันการ Hack ผ่าน Form การเข้าสู่ระบบ



**ภาพที่ 10** โค้ด PDO ป้องกันการ Hack ผ่านการเข้าสู่ระบบ

จากภาพที่ 10 เป็นการแสดงตัวอย่างโค้ดของฟังก์ชั่น PDO PHP Language โดยการ Prepare Variable ไว้ในคำสั่ง SQL เพื่อป้องกันการ Hack ผ่าน SQL Injection

ผลลัพธ์การป้องกัน



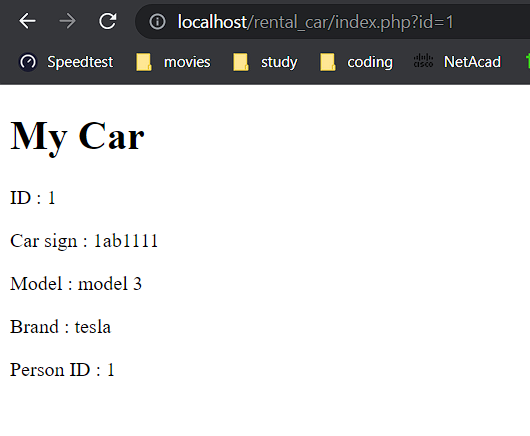
**ภาพที่ 11** ผลลัพธ์การทดสอบการใช้ or ‘1’ = ’1



**ภาพที่ 12** ข้อความแจ้งเตือน

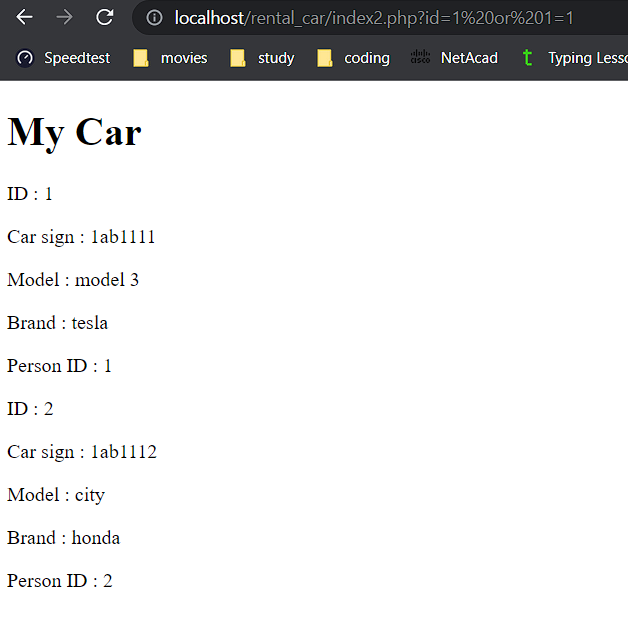
จากภาพที่ 11 แสดงผลลัพธ์การป้องกันการ Hack การเข้าสู่ระบบโดยจะมีการแจ้งเตือนดังภาพที่ 12

ก่อนเขียนโค้ด PDO ฟังก์ชันของ PHP เพื่อป้องกันการ Hack ข้อมูลรายการรถยนต์ ผ่าน URL



**ภาพที่ 13** แสดงข้อมูลรายการรถยนต์เฉพาะของผู้ใช้งาน

จากภาพที่ 13 แสดงรายการรถยนต์เฉพาะของผู้ใช้งานแบบปกติโดยไม่มีการ Hack SQL Injection



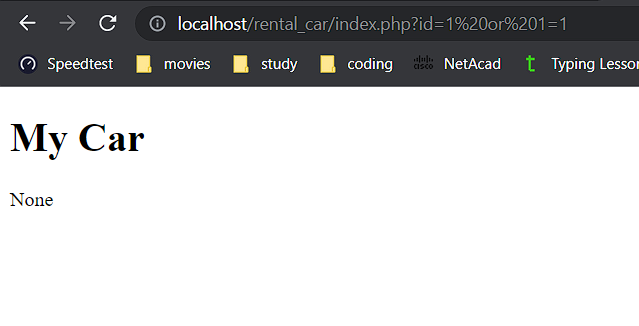
**ภาพที่14** Hack ดูรายการรถยนต์ของผู้อื่น

จากภาพที่ 14 แสดงรายการรถยนต์ทั้งหมดถึงแม้จะไม่ใช่รายการรถยนต์ของผู้ใช้งาน

หลังจากเขียนโค้ดPDO ฟังก์ชันของ PHP เพื่อป้องกันการHackระบบฐานข้อมูลรายการรถยนต์ ผ่าน URL



**ภาพที่ 15** แสดงโค้ด PDO ป้องกันการเข้าถึงข้อมูลรายการรถยนต์ของผู้อื่น



**ภาพที่ 16** แสดงหน้าเพจที่ไม่มีข้อมูลรายการรถยนต์

จากภาพที่ 15 แสดงตัวอย่างโค้ดป้องกันการ hack ผ่าน SQL Injection จากช่อง URL โดยการใช้ฟังก์ชั่น Prepare Variable ไว้ในคำสั่ง SQL แทนการเอาค่าที่รับมาจาก URL ไปใส่ใน SQL ตรงๆ จะได้ผลลัพธ์ดังภาพที่ 16

**สรุปผลวิจัย**

จากการวิจัยการแก้ปัญหาการโดน Hack ด้วยวิธีการแบบ SQL Injection จึงได้นำโค้ด PDO ที่เป็นฟังก์ชันของ PHP มาปรับใช้เพื่อป้องกันการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต ได้ผลสรุปโดยใช้ Test Case ดังนี้

ตาราง Test Case: การเข้าสู่ระบบแบบที่ยังไม่มีการป้องกัน ดังตารางที่ 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Input** |  |  |  |  |
| **Test ID** | **User** | **Password** | **Expected Results** | **Actual Results** | **Accepted Results** |
| TC001 | admin | 123456 | **PASS** | **PASS** | **PASS** |
| TC002 | admin | or‘1=1’ | **FAIL** | **PASS** | **FAIL** |
| TC003 | or‘1=1’ | 123456 | **FAIL** | **PASS** | **FAIL** |
| TC004 | or‘1=1’ | or‘1=1’ | **FAIL** | **PASS** | **FAIL** |

**ตารางที่1** แสดงผลTest Case การเข้าสู่ระบบแบบที่ยังไม่มีการป้องกัน

จากตารางที่ 1 แสดงการเข้าสู่ระบบแบบที่ยังไม่มีการป้องกัน TC001 ใส่ชื่อและรหัสผ่านถูกต้อง จึงสามารถเข้าสู่ระบบได้ , TC002 ใส่ชื่อถูกต้อง ใส่รหัสผ่านไม่ถูกต้องโดยใส่เป็นคำสั่ง or‘1=1’ จึงทำให้เข้าสู่ระบบได้ , TC003 ใส่รหัสผ่านถูกต้อง ใส่ชื่อไม่ถูกต้องโดยใส่เป็นคำสั่ง or‘1=1’ จึงทำให้เข้าสู่ระบบได้ และ TC004 ใส่คำสั่ง or‘1=1’ทั้งในชื่อและรหัสผ่าน จึงทำให้สามารถเข้าสู่ระบบได้

ตาราง Test Case: การเข้าสู่ระบบแบบที่มีการป้องกันแล้ว ดังตารางที่ 2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Input** |  |  |  |  |
| **Test ID** | **User** | **Password** | **Expected Results** | **Actual Results** | **Accepted Results** |
| TC005 | admin | 123456 | **PASS** | **PASS** | **PASS** |
| TC006 | admin | or‘1=1’ | **FAIL** | **FAIL** | **PASS** |
| TC007 | or‘1=1’ | 123456 | **FAIL** | **FAIL** | **PASS** |
| TC008 | or‘1=1’ | or‘1=1’ | **FAIL** | **FAIL** | **PASS** |

**ตารางที่2** แสดงผลTest Case การเข้าสู่ระบบแบบที่มีการป้องกันแล้ว

จากตารางที่ 2 แสดงการเข้าสู่ระบบแบบที่มีการป้องกันแล้ว TC005 ใส่ชื่อและรหัสผ่านถูกต้อง จึงสามารถเข้าสู่ระบบได้ปกติ , TC006 ใส่ชื่อถูกต้อง ใส่รหัสผ่านไม่ถูกต้องโดยใส่เป็นคำสั่ง or‘1=1’ ก็ไม่สามารถเข้าสู่ระบบได้ , TC007 ใส่รหัสผ่านถูกต้อง ใส่ชื่อไม่ถูกต้องโดยใส่เป็นคำสั่ง or‘1=1’ ก็ไม่สามารถเข้าสู่ระบบได้ และ TC008 ใส่คำสั่ง or‘1=1’ทั้งในชื่อและรหัสผ่าน ก็ทำให้ไม่สามารถเข้าสู่ระบบได้

ตาราง Test Case: การเข้าถึงข้อมูลรายการรถยนต์แบบที่ยังไม่มีการป้องกัน ดังตารางที่ 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Test Steps** | **Expected Results** | **Actual Results** | **Accepted Results** |
| TC009 | 1.เข้าสู่ระบบ  2.คลิกปุ่มรายการรถยยนต์ | แสดงรายการรถยนต์เฉพาะของตัวเองเท่านั้น | แสดงรายการรถยนต์เฉพาะของตัวเอง | **PASS** |
| TC010 | 1.เข้าสู่ระบบ  2.คลิกปุ่มรายการรถยยนต์  3.ใส่คำสั่ง or‘1=1’ในช่องURL | แสดงรายการรถยนต์เฉพาะของตัวเองเท่านั้น | แสดงรายการรถยนต์ของผู้อื่นด้วย | **FAIL** |

ตารางที่3 แสดงผล Test Case การเข้าถึงข้อมูลรายการรถยนต์แบบที่ยังไม่มีการป้องกัน

จากตารางที่ 3 แสดงการเข้าถึงข้อมูลรายการรถยนต์แบบที่ยังไม่มีการป้องกันโดย TC009 เข้าสู่ระบบและคลิกดูรายการรถยนต์ของตนเอง แสดงรายการรถยนต์เฉพาะของตัวเอง TC010 เข้าสู่ระบบและคลิกดูรายการรถยนต์ของตนเอง มีการเพิ่มคำสั่ง or ‘1=1’ในช่องURL ระบบต้องแสดงเฉพาะของตัวเองแต่ระบบแสดงรายการรถยนต์ของผู้อื่นด้วย

ตาราง Test Case: การเข้าถึงข้อมูลรายการรถยนต์แบบที่มีการป้องกันแล้ว ดังตารางที่ 4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Test Steps** | **Expected Results** | **Actual Results** | **Accepted Results** |
| TC011 | 1.เข้าสู่ระบบ  2.คลิกปุ่มรายการรถยยนต์ | แสดงรายการรถยนต์เฉพาะของตัวเองเท่านั้น | แสดงรายการรถยนต์เฉพาะของตัวเอง | **PASS** |
| TC012 | 1.เข้าสู่ระบบ  2.คลิกปุ่มรายการรถยยนต์  3.ใส่คำสั่ง or’1=1’ในช่องURL | ไม่แสดงข้อมูลรายการรถยนต์ผู้อื่น | ไม่แสดงรายการรถยนต์ | **PASS** |

**ตารางที่4** Test Case การเข้าถึงข้อมูลรายการรถยนต์แบบที่มีการป้องกันแล้ว

จากตารางที่ 4 แสดงการเข้าถึงข้อมูลรายการรถยนต์แบบที่มีการป้องกันแล้ว โดย TC011 เข้าสู่ระบบและคลิกดูรายการรถยนต์ของตนเอง แสดงรายการรถยนต์เฉพาะของตัวเอง TC012 เข้าสู่ระบบและคลิกดูรายการรถยนต์ของตนเอง มีการเพิ่มคำสั่ง or ‘1=1’ในช่องURL ระบบต้องไม่แสดงรายการรถยนต์ผู้อื่น ผลคือระบบไม่แสดงรายการรถยนต์

ตาราง Test Case: การเข้าถึงข้อมูลรายการเช่ารถยนต์แบบที่ยังไม่มีการป้องกัน ดังตารางที่ 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Test Steps** | **Expected Results** | **Actual Results** | **Accepted Results** |
| TC013 | 1.เข้าสู่ระบบ  2.คลิกปุ่มรายการเช่า | แสดงรายการเช่ารถยนต์เฉพาะของตัวเองเท่านั้น | แสดงรายการเช่ารถยนต์เฉพาะของตัวเอง | **PASS** |
| TC014 | 1.เข้าสู่ระบบ  2.คลิกปุ่มรายการเช่ารถยยนต์  3.ใส่คำสั่ง or’1=1’ในช่องURL | ไม่แสดงข้อมูลรายการเช่ารถยนต์ผู้อื่น | แสดงรายการเช่ารถยนต์ของผู้อื่น | **FAIL** |

**ตารางที่5** แสดงผล Test Case การเข้าถึงข้อมูลรายการเช่ารถยนต์แบบที่ยังไม่มีการป้องกัน

จากตารางที่ 5 แสดงการเข้าถึงข้อมูลรายการเช่ารถยนต์แบบที่ยังไม่มีการป้องกันโดย TC013 เข้าสู่ระบบและคลิกดูรายการเช่ารถยนต์ของตนเอง แสดงรายการเช่ารถยนต์เฉพาะของตัวเอง TC014 เข้าสู่ระบบและคลิกดูรายการเช่ารถยนต์ของตนเอง มีการเพิ่มคำสั่ง or ‘1=1’ในช่องURL ระบบต้องแสดงเฉพาะของตัวเองแต่ระบบแสดงรายการเช่ารถยนต์ของผู้อื่นด้วย

ตาราง Test Case: การเข้าถึงข้อมูลรายการเช่ารถยนต์แบบที่มีการป้องกันแล้ว ดังตารางที่ 6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Test Steps** | **Expected Results** | **Actual Results** | **Accepted Results** |
| TC015 | 1.เข้าสู่ระบบ  2.คลิกปุ่มรายการเช่า | แสดงรายการเช่ารถยนต์เฉพาะของตัวเองเท่านั้น | แสดงรายการเช่ารถยนต์เฉพาะของตัวเอง | **PASS** |
| TC016 | 1.เข้าสู่ระบบ  2.คลิกปุ่มรายการเช่ารถยยนต์  3.ใส่คำสั่ง or’1=1’ในช่องURL | ไม่แสดงข้อมูลรายการเช่ารถยนต์ผู้อื่น | ไม่แสดงรายการเช่ารถยนต์ | **PASS** |

**ตารางที่6** แสดงผล Test Case การเข้าถึงข้อมูลรายการเช่ารถยนต์แบบที่มีการป้องกันแล้ว

จากตารางที่ 6 แสดงการเข้าถึงข้อมูลรายการเช่ารถยนต์แบบที่มีการป้องกันแล้ว โดย TC015 เข้าสู่ระบบและคลิกดูรายการเช่ารถยนต์ของตนเอง แสดงรายการเช่ารถยนต์เฉพาะของตัวเอง TC016 เข้าสู่ระบบและคลิกดูรายการเช่ารถยนต์ของตนเอง มีการเพิ่มคำสั่ง or ‘1=1’ในช่องURL ระบบต้องไม่แสดงรายการเช่ารถยนต์ผู้อื่น ผลคือระบบไม่แสดงรายการเช่ารถยนต์

**5. รายการอ้างอิง**

[1]access rights หรือสิทธิการเข้าถึงคืออะไร, (2560,2 กุมภาพันธ์). Access rights (แอคเซ็ซ ไรท์) หรือสิทธิการเข้าถึง.<https://www.mindphp.com/%E0%B8%84%E0%B8%B9%E0%B9%88%E0%B8%A1%E0%B8%B7%E0%B8%AD/73-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3/3937-access-rights.html>

[2] ณัฐกฤตา โกมลนาค.(2565, 17 กันยายน). การรักษาความมั่นคงปลอดภัยด้านสารสนเทศ สำหรับผู้ดูแลระบบ. มหาวิทยาลัยแม่โจ้. <https://erp.mju.ac.th/acticleDetail.aspx?qid=549>

[3] ความมั่นคงปลอดภัยของระบบสารสนเทศ. (ม.ป.ป.).องค์ประกอบของระบบสารสนเทศกับความมั่นคงปลอดภัย.<https://sites.google.com/site/kanokwant551/khwam-mankhng-plxdphay-khxng-rabb-sarsnthes>

[4] การโจมตีด้วยการฉีด SQLคืออะไรและจะป้องกันได้อย่างไร. (ม.ป.ป.). QUISH. <https://th.quish.tv/what-are-sql-injection-attacks>

[5] แววตา เตชาทวีวรรณ. (2020). สภาพและปัญหาการรักษาความมั่นคงปลอดภัยระบบสารสนเทศของห้องสมุดสถาบันอุดมศึกษาของรัฐ, 13(2)

[6] กิตติศักดิ์ แก้วบุตรดี. (2020). ความมั่นคงปลอดภัยระบบสารสนเทศ(ISO27001: 2013)– มิติใหม่ของการบริหารจัดการโรงพยาบาล, 8(2)

[7] ภุมวุฒิ วิทวัสสำราญกุล. (2563). //แนวทางการพัฒนาระบบรักษาความปลอดภัยข้อมูลสารสนเทศโดยใช้กรอบแนวคิดระบบจัดการความปลอดภัยข้อมูลสารสนเทศ. / สารนิพนธ์/ (วิศวกรรมเว็บและการพัฒนาแอปพลิเคชันบนอุปกรณ์พกพา)./มหาวิทยาลัยธุรกิจบัณฑิตย์//ถ่ายเอกสาร.