แนวปฏิบัติ

การใช้เทคโนโลยี Application Programming Interface (API) ในการให้บริการทางการเงิน

XX มิถุนายน 2565

ธนาคารแห่งประเทศไทย

ร่าง

จัดทำโดย

ฝ่ายเทคโนโลยีทางการเงิน และ

ฝ่ายกำกับและตรวจสอบความเสี่ยงด้านเทคโนโลยีสารสนเทศ

สายนโยบายระบบการชำระเงินและเทคโนโลยีทางการเงิน

ธนาคารแห่งประเทศไทย

ร่วมกับ

ศูนย์ประสานงานการรักษาความมั่นคงปลอดภัยเทคโนโลยีสารสนเทศภาคการธนาคาร (TB-CERT)

สมาคมธนาคารไทย

โทรศัพท์ 0 2283 6574

0 2283 6816

e-mail: FinTechDept@bot.or.th

สารบัญ

หัวข้อ **หน้า**

บทสรุปผู้บริหาร (Executive Summary)

ธนาคารแห่งประเทศไทย (ธปท.) เห็นถึงบทบาทและความสำคัญของเทคโนโลยี Application Programming Interface (API) ที่จะสามารถนำมาช่วยเพิ่มประสิทธิภาพในการแลกเปลี่ยนข้อมูลหรือให้บริการระหว่างผู้ให้บริการทางการเงินให้เป็นอัตโนมัติ รวดเร็ว แม่นยำ เปิดกว้าง และเป็นรากฐานสำคัญของการพัฒนานวัตกรรมทางการเงินและระบบการเงินของประเทศไทย จึงได้ออกแนวปฏิบัติฉบับนี้เพื่อให้ผู้ให้บริการทางการเงินใช้อ้างอิงให้การนำเทคโนโลยีดังกล่าวมาใช้ประโยชน์ได้รับการดูแลความเสี่ยงอย่างรัดกุม เพื่อสร้างความเชื่อมั่นและความมั่นคงปลอดภัยแก่ผู้ใช้บริการทางการเงิน

ขอบเขตของแนวปฏิบัติฉบับนี้ครอบคลุมผู้ให้บริการทางการเงินที่อยู่ภายใต้การกำกับดูแลของ ธปท. ได้แก่ สถาบันการเงิน ผู้ประกอบธุรกิจที่ไม่ใช่สถาบันการเงินที่อยู่ภายใต้การกำกับของ ธปท. และผู้ประกอบธุรกิจตามกฎหมายว่าด้วยระบบการชำระเงิน ที่มีการประยุกต์ใช้เทคโนโลยี API ในการให้บริการทางการเงิน

แนวปฏิบัติฉบับนี้ได้กำหนดหลักการพึงปฏิบัติให้ผู้ให้บริการทางการเงินปฏิบัติตามไว้ 6 หลักการ ประกอบด้วย

1. กลไกการบริหารจัดการและกำกับดูแล API (Governance) เพื่อให้มีการบริหารจัดการและกำกับดูแลการใช้เทคโนโลยี API อย่างเหมาะสมตามระดับความเสี่ยงของลักษณะของข้อมูลหรือบริการตลอดทั้งวงจรชีวิตของการให้บริการหรือใช้บริการ API โดยคำนึงถึงปัจจัยต่าง ๆ ที่เกี่ยวข้อง ภายใต้กรอบการบริหารความเสี่ยงที่ดี และส่งเสริมความเปิดกว้าง การใช้งานระหว่างกันได้ รวมทั้งการพัฒนาต่อยอดในอนาคต

2. แนวทางการบริหารจัดการวงจรชีวิต API (API Lifecycle Management) เพื่อให้มีการบริหารจัดการวงจรชีวิต API (API lifecycle) ตั้งแต่สร้าง พัฒนา เผยแพร่ จัดการ ติดตามการใช้งาน และเลิกใช้งาน อย่างเหมาะสม เพื่อสามารถให้บริการได้อย่างต่อเนื่อง และช่วยลดความเสี่ยงจากการถูกโจมตีผ่านช่องทาง API

3. มาตรฐานด้านความมั่นคงปลอดภัย API (API Security Standard) เพื่อให้มีการบริหารจัดการและควบคุมความเสี่ยงด้านเทคโนโลยีสารสนเทศ รวมทั้งการรักษาความมั่นคงปลอดภัยของระบบและข้อมูลตามมาตรฐานสากลและหลักเกณฑ์ที่เกี่ยวข้องอย่างเหมาะสมตามระดับความเสี่ยงของประเภท API และข้อมูล โดยแบ่งแนวทางควบคุมความมั่นคงปลอดภัยไว้เป็น 7 ด้าน ได้แก่

(1) กระบวนการยืนยันตัวตน (Authentication)

(2) กระบวนการตรวจสอบสิทธิ์การใช้งาน (Authorization)

(3) การรักษาความลับของข้อมูลและการตรวจสอบความถูกต้องของข้อมูล (Data Confidentiality and Integrity)

(4) การรักษาความมั่นคงปลอดภัยด้านการสื่อสาร (Secure Communication)

(5) การพัฒนาโปรแกรมและการกำหนดค่าที่ปลอดภัย (Secure Coding and Configuration)

(6) การจัดเก็บข้อมูลบันทึกเหตุการณ์ และการเฝ้าระวัง (Audit Log and Monitoring)

(7) ความเพียงพอของทรัพยากร (Resource Sufficiency)

4. ความสัมพันธ์ระหว่างผู้ให้บริการ API และผู้ใช้บริการ API (Contractual Relationship) เพื่อให้มีการบริหารจัดการความสัมพันธ์และข้อสัญญาระหว่างกันอย่างเหมาะสมตามความสัมพันธ์ทางธุรกิจภายใต้ข้อสัญญาที่มีอยู่ระหว่างกัน กฎหมาย และหลักเกณฑ์การกำกับดูแลที่เกี่ยวข้องสอดคล้องกับระดับความเสี่ยง รวมทั้งมีการกำหนดบทบาทหน้าที่ความรับผิดชอบระหว่างกันที่ชัดเจน และมีการทบทวนมาตรฐานความมั่นคงปลอดภัยระหว่างกันอย่างสม่ำเสมอ

5. การเปิดเผยข้อมูลการให้บริการ API (API Service Information Disclosure) เพื่อให้มีการเปิดเผยข้อมูลพื้นฐานที่จำเป็นต่อการใช้งาน API อย่างเพียงพอต่อการตัดสินใจเลือกใช้งาน หรือนำ API ไปใช้งานได้อย่างมีประสิทธิภาพ มีช่องทางในการทดสอบให้แก่ผู้ใช้งาน และส่งเสริมให้ผู้ใช้บริการมีสภาวะแวดล้อมที่เอื้ออำนวยให้สามารถนำ API ไปใช้พัฒนานวัตกรรมได้อย่างรวดเร็วยิ่งขึ้น

6. การคุ้มครองผู้ใช้บริการทางการเงิน (Customer Protection) เพื่อให้ผู้ใช้บริการทางการเงินได้รับการคุ้มครองตามกฎหมายและตามสิทธิต่าง ๆ ที่ตนมี ได้รับข้อมูลที่เพียงพอ มีช่องทางในการแจ้งปัญหาและข้อร้องเรียน ได้รับการดูแล และการเยียวยาอย่างเหมาะสมเมื่อได้รับความเสียหายจากการใช้บริการ

แนวปฏิบัติการใช้เทคโนโลยี Application Programming Interface (API) ในการให้บริการทางการเงิน (Guideline for Application Programming Interface Technology Adoption in Financial Services)

1. เหตุผลในการออกแนวปฏิบัติ

ด้วยเทคโนโลยี Application Programming Interface (API) เข้ามามีบทบาทสำคัญในการให้บริการทางการเงินในปัจจุบันมากยิ่งขึ้น เนื่องจากเป็นเทคโนโลยีหนึ่งที่ช่วยให้การเชื่อมต่อระหว่างระบบ เพื่อแลกเปลี่ยนข้อมูลและบริการระหว่างผู้ให้บริการทางการเงินมีประสิทธิภาพมากขึ้น ส่งเสริมการใช้ทรัพยากรที่จำเป็นร่วมกัน ลดเวลาและต้นทุนในการพัฒนาระบบ รวมทั้งสนับสนุนการพัฒนาบริการที่หลากหลายเพื่อตอบสนองความต้องการแก่ผู้ใช้บริการทางการเงินได้ดีขึ้น

อย่างไรก็ตาม นอกจากประโยชน์ที่ได้รับจากการใช้เทคโนโลยี API ในการให้บริการทางการเงินแล้ว การนำ API มาใช้ในการให้บริการทางการเงินอาจก่อให้เกิดความเสี่ยงต่าง ๆ ด้วยเช่นกัน ซึ่งอาจกระทบต่อการดำเนินงานหรือการให้บริการของผู้ให้บริการทางการเงิน โดยเฉพาะอย่างยิ่งในความเสี่ยงด้านเทคโนโลยีสารสนเทศ จึงมีความจำเป็นอย่างยิ่งที่การใช้เทคโนโลยี API ในการให้บริการทางการเงินจะต้องดำเนินการพัฒนาและให้บริการภายใต้กรอบการควบคุมดูแลและการบริหารจัดการที่ดี สอดคล้องตามมาตรฐานสากล เพื่อให้ลดผลกระทบที่อาจเกิดจากความเสี่ยงในด้านต่าง ๆ ที่อาจเกิดขึ้น

ธนาคารแห่งประเทศไทย (ธปท.) ร่วมกับศูนย์ประสานงานการรักษาความมั่นคงปลอดภัยเทคโนโลยีสารสนเทศภาคการธนาคาร (TB-CERT) จึงได้จัดทำแนวปฏิบัติการใช้เทคโนโลยี Application Programming Interface (API) ในการให้บริการทางการเงินฉบับนี้ขึ้น โดยมีวัตถุประสงค์ที่สำคัญได้แก่

(1) กำหนดมาตรฐานในการนำเทคโนโลยี API มาใช้ในการให้บริการทางการเงิน เพื่อเป็นมาตรฐานขั้นต่ำให้ผู้ให้บริการ API มีการประยุกต์ใช้มาตรฐานข้อมูลที่ได้รับการยอมรับ ส่งเสริมใช้งานระหว่างกันได้ (Interoperability) มีการรักษาความมั่นปลอดภัยด้านเทคโนโลยีสารสนเทศ ความมั่นคงปลอดภัยของข้อมูลและความมั่นคงปลอดภัยไซเบอร์ ตามมาตรฐานสากลและกฎเกณฑ์เกี่ยวข้อง รวมทั้งมีการบริหารจัดการ API บริหารจัดการความสัมพันธ์ระหว่างผู้ให้บริการและผู้ใช้บริการ API และความเสี่ยงที่เกี่ยวข้องอย่างเหมาะสม

(2) สร้างสภาพแวดล้อมที่สนับสนุนการต่อยอดนวัตกรรมทางการเงิน ซึ่งจะช่วยลดระยะเวลาและอุปสรรคของนักพัฒนาที่นำ API ไปประยุกต์ใช้ โดยจัดให้มีช่องทางสนับสนุนและเปิดเผยข้อมูลที่จำเป็นแก่ผู้ใช้บริการ API

(3) คุ้มครองผู้ใช้บริการทางการเงินในด้านต่าง ๆ เช่น มีการคุ้มครองข้อมูลส่วนบุคคล มีกระบวนการจัดการปัญหาและการเยียวยาชดเชยความเสียหายที่อาจเกิดขึ้นกับผู้ใช้บริการ มีการให้ความรู้ความเข้าใจ และให้ประสบการณ์การใช้งานที่ดี

2. ขอบเขตการใช้

แนวปฏิบัติฉบับนี้ มีวัตถุประสงค์เพื่อให้ผู้ให้บริการทางการเงินที่อยู่ภายใต้การกำกับดูแลของ ธปท. ได้แก่ สถาบันการเงิน ผู้ประกอบธุรกิจที่ไม่ใช่สถาบันการเงินที่อยู่ภายใต้การกำกับของ ธปท. และผู้ประกอบธุรกิจตามกฎหมายว่าด้วยระบบการชำระเงิน ที่มีการประยุกต์ใช้เทคโนโลยี API ในการให้บริการทางการเงิน

นอกจากนี้ หากผู้ให้บริการทางการเงินเห็นว่าการประยุกต์ใช้เทคโนโลยี API ในการให้บริการทางการเงินที่อยู่ภายใต้ขอบเขตการใช้ของแนวปฏิบัติฉบับนี้มีลักษณะเป็นบริการทางการเงินที่จะพัฒนาไปเป็นโครงสร้างพื้นฐานหรือมาตรฐานกลางสำหรับภาคการเงินไทยที่ผู้ให้บริการทางการเงินจำเป็นต้องทดสอบร่วมกัน ผู้ให้บริการทางการเงินควรพิจารณานำการประยุกต์ใช้ API ในการให้บริการทางการเงินดังกล่าวเข้าทดสอบใน Regulatory Sandbox ตามแนวทางที่กำหนดในแนวปฏิบัติ ธปท. ว่าด้วยแนวทางการเข้าร่วมทดสอบและพัฒนานวัตกรรมที่นำเทคโนโลยีใหม่มาสนับสนุนการให้บริการทางการเงิน

สำหรับการประยุกต์ใช้เทคโนโลยี API ที่ไม่อยู่ภายใต้ขอบเขตการใช้ของแนวปฏิบัติฉบับนี้ ผู้ให้บริการทางการเงินอาจนำข้อกำหนดตามแนวปฏิบัติฉบับนี้ไปประยุกต์ใช้กับกรณีดังกล่าวได้ตามความเหมาะสม เช่น บริการ API ที่ไม่เกี่ยวข้องกับการให้บริการทางการเงิน เป็นต้น

3. คำจำกัดความ

API (Application Programming Interface) หมายถึง ชุดของคำสั่งหรือฟังก์ชันในโปรแกรมคอมพิวเตอร์ใด ๆ ซึ่งได้เตรียมไว้โดยมีข้อกำหนดกฎเกณฑ์จำเพาะ และได้เปิดเป็นช่องทางให้นักพัฒนาสามารถพัฒนาโปรแกรมคอมพิวเตอร์อื่นมาเรียกใช้งาน เพื่อสื่อสาร แลกเปลี่ยนข้อมูล หรือมีปฏิสัมพันธ์ต่าง ๆ ระหว่างกัน

ผู้ให้บริการ API (API Provider) หมายถึง หน่วยงานที่ให้บริการ API ให้แก่ผู้ใช้บริการ API เพื่อให้ผู้ใช้บริการ API สามารถเข้าถึงข้อมูลหรือบริการของตน เช่น เปิดบริการให้เรียกดูข้อมูลอัตราแลกเปลี่ยน ข้อมูลบัญชี ข้อมูล Statement เป็นต้น

ผู้ใช้บริการ API (API Consumer) หมายถึง หน่วยงานที่เรียกใช้ API ของผู้ให้บริการ API เพื่อให้ตนเองหรือผู้ใช้บริการทางการเงินได้ใช้ประโยชน์จากข้อมูลหรือบริการจาก API นั้น เช่น บริษัท Fintech ที่เรียกใช้งาน API ของธนาคาร หรือธนาคารเรียกใช้งาน API จาก Tech Company ภายนอก เป็นต้น

ผู้ให้บริการทางการเงิน หมายถึง สถาบันการเงิน และบริษัทในกลุ่มธุรกิจทางการเงินตามกฎหมายว่าด้วยธุรกิจสถาบันการเงิน ผู้ประกอบธุรกิจที่ไม่ใช่สถาบันการเงินที่อยู่ภายใต้การกำกับของ ธปท. และผู้ประกอบธุรกิจตามกฎหมายว่าด้วยระบบการชำระเงิน ที่อยู่ภายใต้ขอบเขตการใช้แนวปฏิบัติฉบับนี้

ผู้ใช้บริการทางการเงิน (Customer) หมายถึง หมายถึง นิติบุคคล หรือบุคคลธรรมดา ซึ่งได้รับข้อมูลหรือบริการจากผู้ให้บริการทางการเงิน ไม่ว่าจะเป็นการได้รับข้อมูลหรือบริการโดยตรงจากผู้ให้บริการทางการเงินนั้นหรือโดยอ้อมจากผู้ให้บริการทางการเงินรายอื่นซึ่งเรียกใช้บริการ API จากผู้ให้บริการทางการเงินนั้น

API มาตรฐาน (Standardized API) หมายถึง API ที่ถูกพัฒนาขึ้นโดยคำนึงถึงและประยุกต์ใช้เทคโนโลยีการสื่อสารข้อมูล มาตรฐานข้อความ และมาตรการควบคุมความเสี่ยงด้านเทคโนโลยีสารสนเทศที่ได้รับการยอมรับ เพื่อใช้เป็นสื่อกลางร่วมกันระหว่างกลุ่มผู้ให้บริการ API และผู้ใช้บริการ API โดยมีความคาดหวังด้วยว่า API ดังกล่าวจะถูกนำไปใช้โดยผู้ให้บริการ API และผู้ใช้บริการ API รายอื่นในวงกว้างเพิ่มเติมในอนาคต

4. ความเสี่ยงที่สำคัญด้านเทคโนโลยีสารสนเทศที่เกี่ยวข้องกับการนำ API มาใช้

การใช้เทคโนโลยี API ในการให้บริการทางการเงินมีความเสี่ยงที่สำคัญด้านเทคโนโลยีสารสนเทศที่พึงต้องได้รับการบริหารจัดการให้ความเสี่ยงเหล่านั้นอยู่ในระดับที่เหมาะสมกับรูปแบบของธุรกรรมและลักษณะของการใช้งาน โดยมีความเสี่ยงที่สำคัญดังนี้

(1) **การลักลอบเข้าถึงข้อมูลหรือใช้งาน API โดยไม่ได้รับสิทธิ์ (Unauthorized Access)** การออกแบบและพัฒนา API โดยมีกลไกการทำงานของ API ที่ไม่รัดกุม หรือมีมาตรการในการรักษาความมั่นคงปลอดภัยของระบบและเครือข่ายคอมพิวเตอร์ที่ไม่เพียงพอ อาจส่งผลให้ข้อมูลหรือระบบคอมพิวเตอร์ของผู้ให้บริการถูกผู้ไม่ประสงค์ดีเข้าถึง ใช้งาน แก้ไข เปลี่ยนแปลง ทำลายข้อมูล หรือใช้งาน API ได้โดยไม่ได้รับสิทธิ์ ทั้งนี้การลักลอบเข้าถึงข้อมูลหรือใช้งาน API อาจเกิดขึ้นได้จากสาเหตุหลายประการ เช่น

- ความเสี่ยงที่เกิดจากระบบหรือผู้ให้บริการขาดการการตรวจสอบสิทธิ์ผู้ใช้งาน API (Broken Object Level Authorization)
- ความเสี่ยงที่เกิดจากระบบหรือผู้ให้บริการขาดการการตรวจสอบสิทธิ์ฟังก์ชันผู้ใช้งาน API (Broken Function Level Authorization)
- ความเสี่ยงที่เกิดจาก API ยอมรับค่าพารามิเตอร์เกินกว่าที่ควรจะเป็น (Mass Assignment)
- ความเสี่ยงที่เกิดจากความบกพร่องของการตรวจสอบข้อมูลก่อนนำไปประมวลผล ทำให้ระบบนำข้อมูลไปประมวลผลผิดพลาด (Injection)
- ความเสี่ยงที่เกิดจากกระบวนการยืนยันตัวตนผิดพลาด (Broken User Authentication)
- ความเสี่ยงที่ข้อมูลถูกดักจับและเปลี่ยนแปลง (Man in the Middle Attack)

(2) **การเปิดเผยข้อมูลเกินจำเป็น (Excessive Data Exposure)** ความเสี่ยงที่เกิดจากการเปิดเผยข้อมูลที่ไม่จำเป็นมากเกินไป หรือการที่ API ตอบกลับข้อมูลมามากเกินกว่าที่จะต้องนำไปใช้งานจริง ซึ่งอาจเกิดจากการออกแบบและพัฒนา API โดยมีกลไกการทำงานของ API ที่ไม่รัดกุม ทั้งนี้การเปิดเผยข้อมูลเกินจำเป็นอาจเกิดขึ้นได้จากสาเหตุหลายประการ เช่น

- ความเสี่ยงที่เกิดจากการพึ่งพาการคัดกรองข้อมูลเฉพาะที่จำเป็นต่อการใช้งานที่ฝั่ง Client (Insecure Client-Side Data Filtering)
- ความเสี่ยงที่เกิดจากข้อความแสดงข้อผิดพลาด แสดงข้อมูลเกินความจำเป็น (Too Informative Error Message)
- ความเสี่ยงที่เกิดจากมีการจัดเก็บข้อมูลที่มีอ่อนไหวไว้ใน Log (Sensitive Information in Log)

(3) **การให้บริการหยุดชะงัก (Service Disruption)** ความไม่เพียงพอของทรัพยากรของระบบหรือการไม่มีมาตรการในการบริหารจัดการการเรียกใช้งาน API ที่ดีเพียงพอ อาจส่งผลกระทบให้การให้บริการ API หยุดชะงักเป็นการชั่วคราวหรือถาวรได้ หากมีการเรียกใช้มากเกินกว่าที่คาดหมายไว้ ทั้งนี้การหยุดชะงักเกิดได้จากสาเหตุหลายประการ เช่น

- ความเสี่ยงจากการคาดการปริมาณการใช้งาน API ที่ต่ำเกินไปส่งผลให้ระบบหยุดชะงักในการให้บริการ (Underestimated Capacity Planning)
- ความเสี่ยงที่เกิดจากขาดการกำหนดหรือจำกัดการใช้งานทรัพยากรบนระบบ (Lack of Resources & Rate Limiting)
- ความเสี่ยงจากการก่อกวนเครือข่ายด้วยการโจมตีระบบจนไม่สามารถให้บริการได้ (Denial of Service Attack)

(4) **ความเสี่ยงที่อาจเกิดจากการบริหารจัดการวงจรชีวิตที่ไม่รัดกุม (Improper API Lifecycle Management)** การบริหารจัดการวงจรชีวิตของ API ตั้งแต่การสร้าง พัฒนา เผยแพร่ จัดการ ติดตามการใช้งาน และเลิกใช้งานที่ไม่เหมาะสม อาจส่งผลให้เกิดความเสี่ยงที่จะส่งผลกระทบต่อทั้งผู้ให้บริการ API และผู้ใช้บริการ API ในหลายประการ ทั้งนี้ ตัวอย่างของความเสี่ยงที่อาจเกิดขึ้นได้ เช่น

- ความเสี่ยงจากการออกแบบและพัฒนา API ที่มีการใช้งานมาตรฐานด้านความมั่นคงปลอดภัยที่ล้าสมัยหรือมีความแข็งแรงที่ไม่เพียงพอ (Design & Develop)
- ความเสี่ยงจากการไม่มีการจัดทำรายการของ API และ API Specification อย่างเหมาะสม ครบถ้วน และเป็นปัจจุบัน (Inventory Management)
- ความเสี่ยงที่เกิดจากขาดการเก็บบันทึกข้อมูลและการเฝ้าระวังตรวจสอบข้อมูล (Transaction Monitoring)

- ความเสี่ยงจากการไม่มีการบริหารจัดการการเลิกใช้และนำออกอย่างเหมาะสม (Deprecation & Retirement Policy)

(5) **การตั้งค่าความมั่นคงปลอดภัยของระบบที่ผิดพลาด (Security Misconfiguration)** ความเสี่ยงของ API ที่เกิดจากการตั้งค่าอย่างไม่ปลอดภัย หรือไม่ได้เปิดใช้งานฟังก์ชันด้านความมั่นคงปลอดภัย เช่น เปิดใช้งานระบบในส่วนที่ไม่จำเป็น การตั้งค่าระบบที่ไม่สอดคล้องกับ Security Hardening Baseline และการตั้งค่าเปิดการแสดงผลข้อมูลสำหรับ Debug ใน Error Message ของระบบทิ้งไว้ มักเป็นสาเหตุที่ทำให้เกิดความเสี่ยงด้านความมั่นคงปลอดภัยในด้านอื่นได้

(6) **การเชื่อมต่อกับหน่วยงานภายนอกที่มีมาตรการรักษาความมั่นคงปลอดภัยที่ไม่เพียงพอ (Inadequate Security Measures of Third Party)** ในกรณีที่การให้บริการ API หรือใช้บริการ API มีการเชื่อมต่อกับหน่วยงานภายนอก ทั้งที่เป็นผู้ให้บริการภายนอก พันธมิตรทางธุรกิจ หรือผู้ใช้บริการทางการเงิน หากหน่วยงานภายนอกเหล่านั้นมีมาตรการในการรักษาความมั่นคงปลอดภัยด้านเทคโนโลยีสารสนเทศที่ไม่เพียงพอ เช่น ระบบได้รับผลกระทบสืบเนื่องจากปัญหาการไม่ตอบสนองของการเรียกใช้บริการ API หรือหน่วยงานภายนอกดังกล่าวถูกโจมตีและผู้โจมตีใช้ช่องการเชื่อมต่อ API โจมตีผู้ให้บริการ API เป็นต้น

5. หลักการพึงปฏิบัติ

หลักการพึงปฏิบัติที่สำคัญในการพัฒนา API มาตรฐานในการให้บริการทางการเงินมี 6 ข้อ ซึ่งผู้ให้บริการทางการเงินต้องถือปฏิบัติเป็นมาตรฐานขั้นต่ำในการให้บริการทางการเงิน โดยมีผลลัพธ์ที่คาดหวังและแนวทางที่พึงปฏิบัติ ดังนี้

หลักการที่ 1 กลไกการบริหารจัดการและกำกับดูแล API (Governance)

ผลลัพธ์ที่คาดหวัง : การให้บริการหรือใช้บริการ API พึงได้รับการบริหารจัดการและกำกับดูแลอย่างเหมาะสมตามระดับความเสี่ยงของลักษณะของข้อมูลหรือบริการตลอดทั้งวงจรชีวิตของการให้บริการหรือใช้บริการ API โดยคำนึงถึงปัจจัยต่าง ๆ ที่เกี่ยวข้อง ภายใต้กรอบการบริหารความเสี่ยงที่ดี และส่งเสริมความเปิดกว้าง การใช้งานระหว่างกันได้ รวมทั้งการพัฒนาต่อยอดในอนาคต

การปฏิบัติตามหลักการนี้จะช่วยให้มีการบริหารจัดการ API และความเสี่ยงที่เกี่ยวข้องอย่างเหมาะสมตามวัตถุประสงค์ข้อที่ 1 ของแนวปฏิบัติฉบับนี้

แนวทางที่พึงปฏิบัติ

(1) **[ประเมินความเหมาะสม]** ผู้ให้บริการ API และผู้ใช้บริการ API ต้องประเมินความเหมาะสมและวางแผนในการให้บริการหรือใช้บริการ API เพื่อการเข้าถึงข้อมูลหรือบริการทางการเงิน โดยต้องคำนึงถึงความสอดคล้องกับกลยุทธ์ทางธุรกิจ ความเสี่ยงด้านกลยุทธ์ ความเสี่ยงด้านชื่อเสียงควบคู่ระหว่างประโยชน์และผลกระทบที่ตนหรือสังคมจะได้รับ ศักยภาพและความเพียงพอของทรัพยากรด้านเทคโนโลยีสารสนเทศ ประเภทและลักษณะของการให้บริการหรือใช้บริการ API รวมทั้งการปฏิบัติตามกฎหมายและหลักเกณฑ์การกำกับดูแลที่เกี่ยวข้อง

(2) **[คณะทำงานด้าน API]** ผู้ให้บริการ API ต้องมีคณะทำงานซึ่งรับผิดชอบในการกำหนดทิศทางบริหารจัดการ กำกับดูแลการพัฒนาและการให้บริการ API โดยคณะทำงานต้องประกอบด้วยผู้มีส่วนเกี่ยวข้องในด้านต่าง ๆ เช่น ด้านธุรกิจ ด้านเทคโนโลยีสารสนเทศ ด้านการบริหารความเสี่ยง และด้านการปฏิบัติตามกฎหมายและหลักเกณฑ์การกำกับดูแล เป็นต้น ซึ่งอาจเป็นบุคลากรภายในของผู้ให้บริการ API เอง หรือมีบุคลากรภายนอกจากผู้ใช้บริการ API หรือผู้มีส่วนเกี่ยวข้องอื่นเข้าร่วมด้วย ทั้งนี้ หากเป็น API มาตรฐาน คณะทำงานต้องมีคุณลักษณะที่มีความเป็นกลาง เปิดกว้าง และมีกลไกการดำเนินการเพื่อสนองตอบความต้องการของผู้มีส่วนเกี่ยวข้องกลุ่มต่าง ๆ ได้

(3) **[จัดสรรทรัพยากร]** ผู้ให้บริการ API ต้องประเมิน จัดสรร และบริหารจัดการงบประมาณและทรัพยากรที่เพียงพอต่อการพัฒนา ให้บริการ และสนับสนุนการให้บริการ API ทั้งนี้ให้คำนึงถึงลักษณะโดยปกติของการให้บริการ API ที่เป็นแผนงานต่อเนื่องระยะยาวด้วย

(4) **[บริหารจัดการ API Lifecycle]** ผู้ให้บริการ API ต้องมีการบริหารจัดการและกำกับดูแลวงจรชีวิตของ API (API Lifecycle) ซึ่งประกอบด้วย การสร้าง พัฒนา เผยแพร่ จัดการ ติดตามการใช้งาน และเลิกใช้งาน อย่างเหมาะสมและสอดคล้องตามข้อกำหนดในหลักการที่ 2 ของแนวปฏิบัติฉบับนี้

(5) **[บริหารความเสี่ยง]** ผู้ให้บริการ API และผู้ใช้บริการ API ต้องมีการบริหารจัดการและกำกับดูแลความเสี่ยงในด้านต่าง ๆ ในระดับที่สอดคล้องกับลักษณะของข้อมูล บริการ และปัจจัยภาวะแวดล้อมอื่น ๆ โดยมีความเสี่ยงที่สำคัญ เช่น ความเสี่ยงด้านปฏิบัติการ ความเสี่ยงจากการเชื่อมต่อกับหน่วยงานภายนอก ความเสี่ยงด้านกฎหมาย ความเสี่ยงด้านชื่อเสียง ความเสี่ยงด้านสัญญาและข้อตกลง ความเสี่ยงด้านผลกระทบทางการเงิน ความเสี่ยงด้านเทคโนโลยีสารสนเทศ โดยเฉพาะอย่างยิ่งกับความเสี่ยงสำคัญด้านเทคโนโลยีสารสนเทศที่เกี่ยวข้องกับการนำ API มาใช้ตามที่ระบุในหัวข้อที่ 4 ของแนวปฏิบัติฉบับนี้ เป็นต้น

(6) **[บริหารและกำกับดูแลตามระดับความเสี่ยงและลักษณะของบริการ]** การบริหารจัดการและกำกับดูแลวงจรชีวิต API และความเสี่ยงด้านต่าง ๆ ตามที่กล่าวถึงในแนวทางพึงปฏิบัติทั้ง 2 ข้อก่อนหน้านี้ ต้องมีการแบ่งแยกหน้าที่และถ่วงดุลในแต่ละบทบาทหน้าที่ในการควบคุม กำกับ และตรวจสอบ (Three lines of defense) ตามหลักการกำกับดูแลกิจการที่ดี และต้องเป็นไปอย่างมีประสิทธิภาพ รัดกุม และเหมาะสมตามระดับความมีนัยสำคัญของความเสี่ยงและผลกระทบต่อการดำเนินธุรกิจของผู้ให้บริการ API หรือผู้ใช้บริการ API โดยสอดคล้องกับลักษณะของข้อมูล บริการ ปัจจัยภาวะแวดล้อมอื่น ๆ เช่น รูปแบบการดำเนินธุรกิจ ปริมาณธุรกรรม ความซับซ้อนของบริการ เครือข่ายคอมพิวเตอร์ที่ใช้งาน และจำนวนและลักษณะของผู้ใช้บริการ API เป็นต้น รวมทั้งสอดคล้องกับข้อกำหนดในแนวนโยบายฉบับนี้และกฎหมายหรือหลักเกณฑ์การกำกับดูแลที่เกี่ยวข้อง

(7) **[ส่งเสริม Interoperability]** ผู้ให้บริการ API ควรให้ความสำคัญต่อการเปิดกว้างและส่งเสริมให้มีการใช้งานระหว่างกันได้ (Openness and Interoperability) เช่น การให้บริการ API สำหรับโครงสร้างพื้นฐาน หรือมาตรฐานกลางสำหรับภาคการเงินไทย การเข้าเป็นส่วนร่วมในโครงการที่ส่งเสริมการเชื่อมโยงระหว่างกันผ่าน API เป็นต้น เพื่อให้ผู้ใช้บริการ API สามารถเชื่อมต่อและเข้าถึงข้อมูลและบริการได้ในรูปแบบเดียวกันอย่างเปิดกว้าง ไม่มีข้อจำกัดปิดกั้นผู้ใช้บริการ API ในการเลือกใช้งาน API จากผู้ให้บริการ API รายอื่น ซึ่งจะช่วยลดต้นทุนและความซ้ำซ้อนในการลงทุนพัฒนาระบบและการเชื่อมโยงระหว่างกัน เกิดรูปแบบธุรกิจที่หลากหลาย สามารถแข่งขันกันได้อย่างไม่มีอุปสรรคและเป็นธรรม และเลือกผู้ให้บริการได้อย่างเสรี รวมถึงต้องให้ความร่วมมือระหว่างผู้ให้บริการ API ผู้ใช้บริการ API หน่วยงานกำกับดูแล และผู้มีส่วนเกี่ยวข้องอื่น ในการสนับสนุนให้เกิด API มาตรฐานและข้อตกลงทางธุรกิจ (Business rule) ที่เปิดกว้างและใช้งานระหว่างกันได้ และสนับสนุนการใช้งาน API มาตรฐานเหล่านั้น

(8) **[มาตรฐานที่เกี่ยวข้องกับ API (Security & Non-Security)]** ผู้ให้บริการและผู้ใช้บริการ API ต้องมีการจัดทำมาตรฐานอย่างเป็นลายลักษณ์อักษรและได้รับการอนุมัติจากคณะกรรมการหรือผู้บริหารที่ได้รับมอบหมาย โดยครอบคลุมในเรื่องต่าง ๆ ที่เกี่ยวข้องกับ API อย่างน้อย ดังนี้

- **[การเลือก Protocol]** ผู้ให้บริการ API ต้องศึกษา ประเมิน คัดเลือก และกำหนดรูปแบบสถาปัตยกรรม (architectural style) หรือเกณฑ์วิธีของการสื่อสารข้อมูล (protocol) ที่เหมาะสมสำหรับ API เช่น REST (Representational State Transfer) หรือ SOAP (Simple Object Access Protocol) โดยพิจารณาจากปัจจัยต่าง ๆ เช่น ข้อจำกัดของทรัพยากรด้านเทคโนโลยีสารสนเทศที่มีอยู่ ความต้องการทั้งเชิงธุรกิจและเชิงประสิทธิภาพ ความเหมาะสมกับประเภทธุรกรรมและรูปแบบของข้อมูลความสะดวกผู้ใช้บริการ API ความสะดวกต่อการพัฒนาต่อยอดเพิ่มเติมในอนาคต ความแพร่หลายและการยอมรับในกลุ่มผู้ใช้งาน เป็นต้น ให้เหมาะสมตามระดับความเสี่ยงที่ยอมรับได้

  ทั้งนี้ สำหรับ API มาตรฐาน รูปแบบสถาปัตยกรรมหรือเกณฑ์วิธีของการสื่อสารข้อมูลที่กำหนด ต้องมีลักษณะเป็นมาตรฐานสากลที่ได้รับการยอมรับและใช้งานโดยทั่วไป และไม่อยู่ภายใต้ข้อจำกัดที่ทำให้สามารถใช้งานได้กับเพียงระบบ ภาษา แพลตฟอร์ม หรือสภาพแวดล้อมทางคอมพิวเตอร์ เฉพาะอย่าง
- **[การเลือก Data Format]** ผู้ให้บริการ API ต้องศึกษา ประเมิน และคัดเลือกรูปแบบโครงสร้างข้อความในการแลกเปลี่ยนข้อมูล (structured data format for data exchanging) ที่เหมาะสม เช่น JSON (JavaScript Object Notation) และ XML (Extensible Markup Language) โดยพิจารณาจากปัจจัยต่าง ๆ เช่น รูปแบบสถาปัตยกรรมหรือเกณฑ์วิธีของการสื่อสารข้อมูลที่ใช้ มาตรฐานข้อความที่ใช้ ข้อจำกัดของทรัพยากรด้านเทคโนโลยีสารสนเทศที่มีอยู่ แนวทางการใช้งานในปัจจุบันของผู้มีส่วนเกี่ยวข้องอื่นในภาคอุตสาหกรรม ระดับของความซับซ้อนของข้อมูล และความสามารถเฉพาะตัวของรูปแบบโครงสร้างข้อความในการแลกเปลี่ยนข้อมูลแต่ละประเภท
- **[การเลือกแนวทางการจัดทำ API Specification]** ผู้ให้บริการ API ต้องศึกษา ประเมิน และคัดเลือกมาตรฐานเพื่อใช้จัดทำเอกสาร API Specification สำหรับพัฒนา และเผยแพร่ให้ใช้งาน โดยคำนึงถึงและปฏิบัติตามมาตรฐานสากลหรือแนวปฏิบัติที่ดีที่เหมาะสมกับรูปแบบสถาปัตยกรรมหรือเกณฑ์วิธีของการสื่อสารข้อมูลที่ใช้ ตัวอย่างของมาตรฐานสากลหรือแนวปฏิบัติที่ดีที่อาจใช้อ้างอิงสำหรับการเผยแพร่ API Specification เช่น มาตรฐาน OAS (OpenAPI Specification) RESTful API Modelling Language (RAML) WSDL

(Web Service Definition Language) เป็นต้น รวมถึงควบคุมการเข้าถึง API Specification ให้สอดคล้องกับวัตถุประสงค์ของการให้บริการ API

- **[มาตรฐาน API Security Standard ขั้นต่ำ]** ผู้ให้บริการ API และผู้ใช้บริการ API ต้องศึกษาและจัดทำมาตรฐานขั้นต่ำการรักษาความมั่นคงปลอดภัยที่เกี่ยวข้องกับ API สอดคล้องตามหลักการที่ 3 ของแนวปฏิบัติฉบับนี้ เช่น มาตรฐานการยืนยันตัวตนและการกำหนดสิทธิ์ มาตรฐานการเข้ารหัสข้อมูลและความยาวกุญแจ เป็นต้น ตามความเสี่ยงและรูปแบบที่ให้หรือใช้บริการ

ตัวอย่างการเลือกใช้มาตรฐานหรือแนวปฏิบัติที่ดีสำหรับอ้างอิง เช่น มาตรฐาน ISO 23029[1] สำหรับการจัดทำมาตรฐาน API รวมทั้ง ควรพิจารณาเลือกใช้โดยระมัดระวังไม่นำมาตรฐานหรือแนวปฏิบัติที่ล้าสมัยหรือเคยได้รับการพิสูจน์แล้วว่ามีช่องโหว่ด้านความมั่นคงปลอดภัยมาใช้งาน

ทั้งนี้ มาตรฐานที่เกี่ยวข้องกับ API อาจพิจารณาอ้างอิงมาตรฐานการรักษาความมั่นคงปลอดภัยด้านเทคโนโลยีสารสนเทศเดิมหรือเอกสารอื่น ๆ ที่เกี่ยวข้องและมีการใช้งานอยู่ โดยขยายขอบเขตให้ครอบคลุมการให้และใช้บริการ API หรืออาจปรับปรุงขึ้นใหม่เพื่อบริหารจัดการเฉพาะที่เกี่ยวข้องกับ API

หลักการที่ 2 แนวทางการบริหารจัดการวงจรชีวิต API (API Lifecycle Management)

ผลลัพธ์ที่คาดหวัง : การให้บริการหรือใช้บริการ API พึงได้รับการบริหารจัดการวงจรชีวิต API (API lifecycle) ตั้งแต่สร้าง พัฒนา เผยแพร่ จัดการ ติดตามการใช้งาน และเลิกใช้งาน อย่างเหมาะสม เพื่อสามารถให้บริการได้อย่างต่อเนื่อง และช่วยลดความเสี่ยงจากการถูกโจมตีผ่านช่องทาง API

การปฏิบัติตามหลักการนี้จะช่วยให้มีการบริหารจัดการ API และความเสี่ยงที่เกี่ยวข้องอย่างเหมาะสม และมีการประยุกต์ใช้มาตรฐานข้อมูลที่ได้รับการยอมรับตามวัตถุประสงค์ข้อที่ 1 ของแนวปฏิบัติฉบับนี้ และช่วยปิดความเสี่ยงในด้านการเปิดเผยข้อมูลเกินจำเป็น (Excessive Data Exposure) การให้บริการหยุดชะงัก (Service Disruption) ความเสี่ยงที่อาจเกิดจากการบริหารจัดการวงจรชีวิตที่ไม่รัดกุม (Improper API Lifecycle Management) และการเชื่อมต่อกับหน่วยงานภายนอกที่มีมาตรการรักษาความมั่นคงปลอดภัยที่ไม่เพียงพอ (Inadequate Security Measures of Third Party)

แนวทางที่พึงปฏิบัติ :

[1] ISO/TS 23029 - Web-service-based application programming interface (WAPI) in financial services

สร้าง (Create) ผู้ให้บริการ API ต้องมีแนวทางในการออกแบบและเตรียมการในการพัฒนาโดยอ้างอิงตามมาตรฐานสากล แนวปฏิบัติที่ดี และมาตรฐานด้าน API ที่กำหนด ซึ่งต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

(1) [ออกแบบ API] ผู้ให้บริการ API ต้องศึกษาและออกแบบ โดยคำนึงถึงความต้องการทางธุรกิจและความมั่นคงปลอดภัยด้าน IT (secured by design) รวมทั้ง การคุ้มครองข้อมูลส่วนบุคคล (privacy by design) ในกรณีที่มีการรับส่งข้อมูลส่วนบุคคล ซึ่งต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

- การออกแบบอินเตอร์เฟสของ API (interface design) เช่น การออกแบบฟังก์ชัน และการกำหนดพารามิเตอร์ สำหรับให้บริการ API เป็นต้น รวมถึงการตั้งชื่อ API และข้อมูลที่เกี่ยวข้อง (naming convention) ตัวอย่างของมาตรฐานสากลหรือแนวปฏิบัติที่ดีที่อาจใช้อ้างอิง เช่น มาตรฐาน BIAN (Banking Industry Architecture Network) มาตรฐาน ISO 23029 เป็นต้น
- การเลือกมาตรฐานข้อความ (message standard) ที่เหมาะสมสำหรับ API ซึ่งควรเลือกใช้มาตรฐานข้อความในระดับสากล มาตรฐานข้อความของประเทศ หรือมาตรฐานข้อความในกลุ่มอุตสาหกรรม แทนการกำหนดมาตรฐานข้อความใหม่เพื่อใช้งานเฉพาะ โดยพิจารณาจากปัจจัยต่าง ๆ เช่น ความเหมาะสมกับประเภทธุรกรรมและรูปแบบของข้อมูล ความต้องการเชิงธุรกิจ และการพัฒนาต่อยอดเพิ่มเติมในอนาคต เป็นต้น ตัวอย่างของมาตรฐานสากลหรือแนวปฏิบัติที่ดีที่อาจใช้อ้างอิง เช่น มาตรฐาน ISO 20022[2] เป็นต้น ทั้งนี้ หากผู้ให้บริการ API พบว่าไม่มีมาตรฐานข้อความใดที่เหมาะสม อาจกำหนดมาตรฐานข้อความขึ้นใหม่เพื่อใช้เองได้
- กำหนดข้อความแสดงข้อผิดพลาด (error message) ที่สื่อความหมายให้บุคคลทั่วไปสามารถเข้าใจได้ สอดคล้องตรงตามข้อผิดพลาดที่เกิดขึ้นจริง เพื่อให้ผู้ใช้บริการทางการเงินสามารถเข้าใจปัญหาในเบื้องต้น และช่วยให้ผู้ให้บริการ API หรือผู้ใช้บริการ API สามารถให้ความช่วยเหลือได้อย่างตรงจุด รวมทั้ง ต้องคำนึงถึงการไม่แสดงข้อมูลเกินความจำเป็นที่อาจถูกใช้ในการหาประโยชน์โดยมิชอบ และอาจใช้เป็นข้อมูลในการการบุกรุกโจมตี

(2) [ออกแบบกระบวนการให้บริการ] ในกรณีที่เป็น API มาตรฐาน ผู้ให้บริการ API ต้องออกแบบกระบวนการให้บริการที่เกี่ยวข้องกับการออกแบบ API มาตรฐาน โดยคำนึงถึงความมั่นคงปลอดภัย

[2] ISO 20022 - Financial services – Universal financial industry message scheme

ประสบการณ์การใช้บริการทางการเงินที่ง่าย สะดวก รวดเร็ว และเหมาะสมกับประเภท ระดับความเสี่ยง ระดับความเชี่ยวชาญของผู้ใช้บริการทางการเงิน

ทั้งนี้ ควรออกแบบให้กระบวนการให้บริการนั้นมีขั้นตอนการใช้งานเท่าที่จำเป็น และไม่ซ้ำซ้อนกัน โดยเฉพาะอย่างยิ่ง ขั้นตอนการยืนยันตัวตน และการขอความยินยอมจากผู้ใช้บริการทางการเงิน

พัฒนาและเผยแพร่ (Develop & Publish) ผู้ให้บริการ API ต้องมีแนวทางในการพัฒนา ทดสอบ และเผยแพร่ API รวมทั้ง API Specification เพื่อให้มั่นใจว่ามีความถูกต้อง มั่นคงปลอดภัย เชื่อถือได้ พร้อมใช้งาน และมีความยืดหยุ่นเพียงพอที่จะรองรับการปรับปรุงเปลี่ยนแปลงในอนาคต ซึ่งต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

(1) [การพัฒนาและการทดสอบ] ผู้ให้บริการ API ต้องพัฒนาระบบเพื่อให้บริการ API สอดคล้องตามมาตรฐานและระเบียบวิธีปฏิบัติในการออกแบบและพัฒนาระบบอย่างปลอดภัย (secure coding) ครอบคลุมทั้งด้านการรักษาความมั่นคงปลอดภัย ด้านประสิทธิภาพการทำงานและการจัดการ ซึ่งสามารถอ้างอิงจากมาตรฐานหรือแนวปฏิบัติที่ดี เช่น OWASP Secure Coding Practices เป็นต้น ต้องสอบทานคำสั่งเขียนโปรแกรม (source code review) และต้องทดสอบระบบก่อนการใช้งานจริง เช่น ทดสอบการทำงานของแต่ละระบบ (unit test) ทดสอบการทำงานร่วมกันของระบบต่าง ๆ (system integration test) ทดสอบความพร้อมใช้งานตามกระบวนการและความต้องการของผู้ใช้งาน (user acceptance test) ทดสอบประสิทธิภาพการทำงาน (performance test) ทดสอบความมั่นคงปลอดภัยของระบบ (security test) เช่น การประเมินช่องโหว่ (vulnerability assessment) ทดสอบเจาะระบบ (penetration test) เป็นต้น

หากเป็นการให้บริการ API สำหรับโครงสร้างพื้นฐาน หรือมาตรฐานกลางสำหรับภาคการเงินไทย ต้องจัดให้มีการดำเนินการทดสอบร่วมจากผู้มีส่วนเกี่ยวข้องก่อนเปิดให้บริการ

(2) [การให้บริการจริง] ผู้ให้บริการ API ต้องมีการบริหารจัดการในด้านต่าง ๆ ที่เกี่ยวข้องกับ API ก่อนเปิดให้บริการหรือเปิดการเชื่อมต่อจริง ซึ่งต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

- การประเมินความเสี่ยงจากการเชื่อมต่อกับหน่วยงานภายนอก ครองคลุมทั้งในกรณีที่เป็นการให้หรือใช้บริการ API ก่อนการเชื่อมต่อจริง
- ทดสอบด้านการรักษาความมั่นคงปลอดภัยของการเชื่อมต่อ API ก่อนให้บริการจริง
- การบริหารจัดการและควบคุมการเปลี่ยนแปลง (change management) ที่รัดกุมเพียงพอสำหรับนำระบบขึ้นใช้งานจริง (system deployment) หรือตั้งค่าระบบ (system configuration) เพื่อเชื่อมต่อระบบที่ให้บริการจริง (production) กับผู้ให้หรือใช้บริการ API
- การบริหารจัดการทะเบียนทรัพย์สิน API (API inventory management) การควบคุมเวอร์ชัน API (API version control) รวมทั้ง API Specification สำหรับพัฒนาและเผยแพร่

ให้ใช้งาน โดยต้องมีการสอบทานทะเบียนทรัพย์สิน API ให้เป็นปัจจุบัน และสอดคล้องกับการให้หรือใช้บริการจริง รวมทั้งควบคุมเวอร์ชันของ API และเอกสารที่เผยแพร่ให้ใช้งาน ให้ถูกต้องและสอดคล้องกัน ตัวอย่างการจัดเก็บทะเบียนทรัพย์สิน API เช่น ชื่อบริการ คำอธิบาย วัตถุประสงค์ เวอร์ชัน ประเภท รูปแบบการเชื่อมต่อ URI/URL path รูปแบบการยืนยันตัวตน ระบบงานที่เกี่ยวข้อง หน่วยงานที่เป็นเจ้าของ หน่วยงานที่ใช้งาน เป็นต้น รวมถึงการจำกัดสิทธิ์ในการเข้าถึง API inventory และ API Specification ให้เฉพาะผู้ที่ได้รับอนุญาตเท่านั้น

- การบริการจัดการ API Key/Token ที่เกี่ยวข้องกับการให้หรือใช้บริการ API (API Key/Token management) ครอบคลุมตั้งแต่การสร้าง การแจกจ่าย จัดเก็บ การใช้งาน การต่ออายุ การเพิกถอน และการสอบทาน โดยต้องคำนึงถึงความไม่ซ้ำกันในการสร้าง ความมั่นคงปลอดภัยของการการแจกจ่ายและจัดเก็บ การตรวจสอบความถูกต้องทุกครั้งก่อนดำเนินการให้บริการ การกำหนดระยะเวลาหมดอายุที่เหมาะสม รวมทั้ง หากมีการใช้งานกุญแจเข้ารหัส ต้องปฏิบัติให้สอดคล้องกับนโยบายหรือมาตรฐานการบริหารจัดการกุญแจเข้ารหัส

ทั้งนี้ อาจพิจารณานำเอาเครื่องมือมาช่วยให้กระบวนการพัฒนา ทดสอบ และให้บริการจริงเป็นไปอย่างอัตโนมัติ (automation) รวมถึงอาจนำเครื่องมือเฉพาะของการทดสอบ API มาใช้เพื่อช่วยลดระยะเวลาในการทดสอบและช่วยลดความผิดพลาดจากการปฏิบัติงาน

จัดการและติดตาม (Operate and Monitor) ผู้ให้บริการ API และผู้ใช้บริการ API ต้องมีแนวทางในจัดการ ติดตามภัยคุกคาม ประสิทธิภาพและความพร้อมใช้งานของการให้หรือใช้บริการ API รวมทั้ง มีการบริหารจัดการเหตุการณ์ผิดปกติและปัญหาที่เกิดจากการใช้เทคโนโลยี (IT incident and problem management) อย่างเหมาะสม เพื่อให้สามารถตรวจจับ ป้องกัน และรับมือเหตุการณ์ผิดปกติได้อย่างทันท่วงที ซึ่งต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

(1) [การจัดการและติดตามด้านความมั่นคงปลอดภัย] ผู้ให้บริการ API ต้องมีแนวทางและการติดตามด้านความมั่นคงปลอดภัยครอบคลุมอย่างน้อยในเรื่องดังต่อไปนี้

- การติดตามดูแลระบบและเฝ้าระวังภัยคุกคาม (security monitoring) โดยมีกระบวนการหรือเครื่องมือในการตรวจจับเหตุการณ์ผิดปกติ หรือภัยคุกคามที่มีผลกระทบต่อความมั่นคงปลอดภัยของระบบที่ให้บริการ API เพื่อให้สามารถตรวจจับ ป้องกัน และรับมือเหตุการณ์ผิดปกติและภัยคุกคามได้ รวมทั้ง ต้องมีแนวทางในการตัดการเชื่อมต่อเพื่อใช้ตัดการทำงาน

ชั่วคราว และลดผลกระทบเมื่อได้ทราบหรือตรวจสอบพบการโจมตี หรือภัยคุกคามเกิดขึ้นกับระบบทั้งฝั่งของผู้ให้บริการและผู้ใช้บริการ API

- การติดตามความก้าวหน้าทางเทคโนโลยีที่เกี่ยวข้องกับการให้บริการ และรูปแบบของการเกิดเหตุภัยคุกคามผ่านช่องทาง API เพื่อนำมาปรับปรุงมาตรการในการรักษาความมั่นคงปลอดภัยในการให้บริการ API ได้อย่างเหมาะสม เช่น OWASP API Security Top 10 Common Weakness Enumeration (CWE) Top 25 หน่วยงานที่ให้บริการ Cyber Threat Intelligence เป็นต้น
- สำหรับกรณีที่เป็นผู้ใช้บริการ API ต้องมีการติดตามและปรับปรุงเวอร์ชันของ API ให้เป็นเวอร์ชันที่มีการสนับสนุนและมีความมั่นคงปลอดภัยอย่างสม่ำเสมอ เช่น ติดตามข้อมูลจากเว็บไซต์ทางการหรือช่องทางสื่อสารอื่น ๆ ของผู้ให้บริการ เป็นต้น ทั้งนี้ ผู้ใช้บริการ API ควรประเมินความเสี่ยงและความจำเป็นก่อนปรับปรุงเวอร์ชัน รวมทั้ง ต้องมีการทดสอบก่อนนำไปใช้งานจริง
- ตรวจหา API ที่ถูกเปิดให้บริการโดยไม่ได้รับอนุญาต ที่อาจจะก่อให้เกิดความเสี่ยงในการเข้าถึงข้อมูล หรืออาจจะถูกใช้เป็นช่องทางในการโจมตีทางไซเบอร์ โดยควรมีการตรวจหาอย่างสม่ำเสมอ

(2) [การจัดการติดตามด้านประสิทธิภาพและความพร้อมใช้งาน] ผู้ให้บริการ API ต้องมีแนวทางและการติดตามด้านประสิทธิภาพและความพร้อมใช้งานของการให้หรือใช้บริการ API ครอบคลุมอย่างน้อยในเรื่องดังต่อไปนี้

- การบริหารจัดการขีดความสามารถ (Capacity management) และการบริหารจัดการความพร้อมใช้งาน (Availability management) และการติดตามประสิทธิภาพ (Performance monitoring) ของระบบที่ให้หรือใช้บริการ API เช่น สถานะการให้บริการ ความเร็วในการตอบสนอง จำนวนครั้งที่มีการใช้งาน ข้อมูลจำนวนการเรียกใช้งาน API ที่ผิดพลาด เป็นต้น เพื่อให้สามารถประเมิน ติดตามเฝ้าระวัง และวางแผนการใช้งานทรัพยากรด้านเทคโนโลยีสารสนเทศของระบบ รวมทั้ง ควรกำหนดตัวชี้วัดการใช้ทรัพยากรด้านเทคโนโลยีสารสนเทศ (threshold และ trigger) ในด้านต่าง ๆ โดยอาจพิจารณาจากปริมาณการใช้งานจริง ปริมาณทรัพยากรระบบ และสามารถปรับปรุงค่าดังกล่าวให้สอดคล้องตามสถานการณ์ในแต่ละช่วงตามความเหมาะสม เพื่อให้มีการแจ้งเตือนไปยังผู้เกี่ยวข้องและหาทางรับมืออย่างทันท่วงที

- การบริหารจัดการอัตราการเรียกใช้งาน (Rate limiting / Throttling management) ของระบบที่ให้บริการ API เพื่อป้องกันปัญหาการเรียกใช้งานในปริมาณที่มากจนอาจส่งผลกระทบต่อความพร้อมใช้งาน (availability) ระบบของผู้ให้บริการ API โดยต้องกำหนด ติดตามและปรับปรุง อัตราการเรียกใช้งาน ที่ใช้ระบุปริมาณการเรียกใช้งาน API ภายในช่วงระยะเวลา ให้มีความสอดคล้องกับธุรกิจและเหมาะสมปริมาณทรัพยากรด้านเทคโนโลยีสารสนเทศที่ให้บริการ
- ต้องมีการจัดทำแผนรองรับการดำเนินธุรกิจอย่างต่อเนื่อง (Business Continuity Plan : BCP) และแผนฉุกเฉินด้านเทคโนโลยีสารสนเทศ (IT Disaster Recovery Plan : IT DRP) ของระบบงานที่ให้บริการ API ตามระดับความสำคัญของระบบงาน
- สำหรับกรณีที่เป็นผู้ใช้บริการ API ต้องมีแนวทางในการตัดการเชื่อมต่อ (Circuit breaker) เพื่อใช้ตัดการทำงาน และลดผลกระทบที่สืบเนื่องจากปัญหาในการเรียกใช้บริการ API เช่น ตัดการเชื่อมการเรียกใช้งาน API จากภายนอก ในกรณีที่ใช้งานไม่ได้และยังคงค้างการเชื่อมต่อจนอาจส่งผลกระทบไปยังระบบอื่น เป็นต้น โดยต้องกำหนด ปรับปรุง ติดตามดูแลเงื่อนไขการตัดการเชื่อมต่อและการกลับมาเปิดให้บริการเมื่อสถานการณ์กลับมาสู่ภาวะปกติ โดยคำนึงถึงความมั่นคงปลอดภัย ความสอดคล้องกับธุรกิจและเหมาะสมกับการใช้งาน

เลิกใช้และนำออก (Deprecate and Retire) ผู้ให้บริการ API ต้องมีแนวทางในการบริหารจัดการ API และเอกสารประกอบการใช้งาน API เมื่อพบว่า API ดังกล่าวหมดอายุการใช้งานหรือไม่เหมาะสมที่จะเปิดให้บริการ ซึ่งอาจมาจากสาเหตุด้านความมั่นคงปลอดภัย ประสิทธิภาพในการให้บริการ หรือการเปลี่ยนแปลงทางธุรกิจ ซึ่งต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

(1) [การเลิกให้บริการ Deprecation] ผู้ให้บริการ API ต้องมีแนวทางในการจัดการยกเลิกให้บริการ (deprecation) API ทั้งหมดหรือบางส่วน และต้องไม่ขัดต่อสัญญาการให้บริการ API ของผู้ให้บริการ ซึ่งต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

- แจ้งความจำนงในการเลิกให้บริการผ่านการสื่อสารอย่างชัดเจน และต้องระบุวันเวลาที่จะยกเลิกให้บริการล่วงหน้า โดยมีระยะเวลาอย่างเหมาะสม เช่น ประกาศยกเลิกการให้บริการ 3 – 6 เดือนล่วงหน้า เป็นต้น เพื่อให้ผู้ใช้บริการ API มีเวลาเพียงพอในการปรับเปลี่ยนระบบงาน รวมทั้ง ต้องระบุผลกระทบที่อาจเกิดขึ้น รวมทั้งช่องทางสำหรับติดต่อผู้ให้บริการ API เพื่อขอการสนับสนุนหากเกิดปัญหา
- จัดเตรียมเวอร์ชันใหม่สำหรับทดแทน โดยอาจเปิดให้บริการแบบคู่ขนานระหว่าง API ที่จะเลิกให้บริการและ API ที่จะให้ใช้ทดแทน ซึ่งควรคำนึงถึงเรื่องความเข้ากันได้กับระบบ

ที่ใช้บริการเวอร์ชันเก่า (Backward Compatibility) หรือควรเสนอสิ่งอื่นให้ผู้ใช้บริการ API พิจารณานำไปใช้เพื่อทดแทน API ที่จะเลิกให้บริการหากไม่ได้มีเวอร์ชันใหม่

(2) [การนำออกจากระบบ (Retirement)] ผู้ให้บริการ API ต้องมีแนวทางที่เหมาะสมในการนำ API ทั้งหมดหรือบางส่วนออกจากระบบ เพื่อลดภาระในการบริหารจัดการเวอร์ชันและป้องกันภัยคุกคามที่อาจเกิดจากการเรียกใช้ API เวอร์ชันเก่าที่ไม่ปลอดภัย เช่น การนำเอา API และเอกสารที่เกี่ยวข้องของ API หลังจากสิ้นสุดสถานะยกเลิกให้บริการ (deprecated) ไปแล้ว 6 เดือนออกจากระบบ เป็นต้น

ทั้งนี้ กระบวนการบริหารจัดการต่าง ๆ ตลอดวงจรชีวิต API อาจพิจารณาอ้างอิงกระบวนการเดิมที่ใช้งานอยู่ โดยขยายขอบเขตให้ครอบคลุมการให้และใช้บริการ API หรือปรับปรุงเป็นกระบวนการใหม่ขึ้นมาเพื่อบริหารจัดการเฉพาะที่เกี่ยวข้องกับ API

หลักการที่ 3 มาตรฐานด้านความมั่นคงปลอดภัย API (API Security Standard)

ผลลัพธ์ที่คาดหวัง : การให้บริการ API ต้องได้รับการบริหารจัดการและควบคุมความเสี่ยงด้านเทคโนโลยีสารสนเทศ รวมทั้งการรักษาความมั่นคงปลอดภัยของระบบและข้อมูลตามมาตรฐานสากลและหลักเกณฑ์ที่เกี่ยวข้องอย่างเหมาะสมตามระดับความเสี่ยงของประเภท API และข้อมูล สำหรับกรอบการกำหนดมาตรฐานในที่นี้เป็นมาตรฐานด้านความมั่นคงปลอดภัยของ API ที่ใช้บริการบน HTTP protocol สำหรับบางบริการ API ที่ให้บริการบน protocol อื่น ซึ่งไม่สามารถจัดให้มีตัวควบคุมตามแนวทางพึงปฏิบัติในหลักการนี้ ควรพิจารณานำตัวควบคุมอื่นที่เทียบเท่ามาทดแทนตามที่เทคโนโลยีของบริการนั้นจะสามารถทำได้ โดยการประเมินระดับความเสี่ยงตั้งต้นของ API และการกำหนดมาตรฐานการควบคุมด้านความมั่นคงปลอดภัยสำหรับการใช้เทคโนโลยี API เพื่อให้บริการทางการเงิน มีรายละเอียดระบุในภาคผนวก 2

การปฏิบัติตามหลักการนี้จะช่วยให้มีการรักษาความมั่นปลอดภัยด้านเทคโนโลยีสารสนเทศ ความมั่นคงปลอดภัยของข้อมูล และความมั่นคงปลอดภัยไซเบอร์ ตามมาตรฐานสากลตามวัตถุประสงค์ข้อที่ 1 ของแนวปฏิบัติฉบับนี้ และช่วยปิดความเสี่ยงในด้านการลักลอบเข้าถึงข้อมูลหรือใช้งาน API โดยไม่ได้รับสิทธิ์ (Unauthorized Access) การเปิดเผยข้อมูลเกินจำเป็น (Excessive Data Exposure) การให้บริการหยุดชะงัก (Service Disruption) และการตั้งค่าความมั่นคงปลอดภัยของระบบที่ผิดพลาด (Security Misconfiguration)

แนวทางที่พึงปฏิบัติ :

(1) [กระบวนการยืนยันตัวตน (Authentication)] ผู้ให้บริการ API ต้องจัดให้มีกระบวนการในการยืนยันตัวตนของผู้ใช้บริการ API ที่รัดกุม ปลอดภัย เหมาะสมกับลักษณะและความเสี่ยงของบริการที่ใช้ เพื่อควบคุมการเข้าถึงข้อมูลของผู้ใช้บริการ API และป้องกันการเปิดช่องโดยไม่ตั้งใจให้ผู้อื่นเข้าถึงข้อมูลโดย

ไม่ได้รับอนุญาต โดยผู้ให้บริการ API ต้องมีกระบวนการในการบริหารจัดการเกี่ยวกับการยืนยันตัวตน เช่น การสร้าง บริหารจัดการ การแสดงผล เพิกถอน ต่ออายุ และการให้สิทธิ์แก่ปัจจัยและข้อมูลที่ใช้ในการยืนยันตัวตน อย่างเหมาะสม ปลอดภัย เพื่อป้องกันไม่ให้ข้อมูลและปัจจัยในการยืนยันตัวตนถูกแก้ไข เปลี่ยนแปลงหรือผู้อื่นแอบอ้างใช้งานได้ ในที่นี้จะกล่าวรวมถึงการยืนยันตัวตนทั้งด้านผู้ใช้บริการ และด้านระบบผู้ให้บริการ

1.1. [กระบวนการยืนยันตัวตนทางด้านผู้ใช้บริการ หรือ Client authentication] การยืนยันตัวตนทางฝั่งผู้ใช้บริการหรือระบบต้นทางที่ต้องการเชื่อมต่อไปยังอีกระบบนึง เพื่อขอใช้บริการต่าง ๆ แนวทางที่พัฒนาใช้กันส่วนใหญ่มีวิธีดังนี้

- การยืนยันตัวตนโดยใช้ Username และ Password คือรูปแบบการยืนยันตัวตนที่ง่ายที่สุด อย่างไรก็ตามการใช้ Username และ Password มีจุดอ่อนคือ การบริหารจัดการให้มีความมั่นคงปลอดภัย ตั้งแต่การสร้างรหัสผ่านต้องคาดเดาได้ยาก และการเปลี่ยนรหัสผ่านเมื่อพบว่าข้อมูลรหัสผ่านมีความเสี่ยงจากเหตุการณ์ข้อมูลรั่วไหล และการป้องกันความลับของข้อมูลรหัสผ่านขณะส่ง ด้วยเหตุนี้จึงต้องใช้ร่วมกับ HTTPS เพื่อให้เกิดการรักษาความมั่นคงปลอดภัยของข้อมูลที่ใช้ในการยืนยันตัวตน
- การยืนยันตัวตนโดยใช้ Token-based คือรูปแบบการยืนยันตัวตนที่ใช้รหัส Token แทนการส่งด้วยข้อมูล Username และ Password โดยตรง เช่น JWT (JSON Web Tokens), API key เป็นต้น ซึ่งข้อดีของวิธีนี้คือ การยืนยันตัวตนมีความมั่นคงปลอดภัยมากขึ้น เนื่องจากระบบจะมีการสร้างค่า Token สำหรับใช้งานแต่ละครั้ง และสามารถกำหนดการใช้งานในระยะเวลาที่จำกัด

ทั้งนี้ ผู้ให้บริการ API ต้องดูแลข้อมูลที่ใช้ยืนยันตัวตนสำหรับการใช้งาน API ของผู้ใช้บริการ API เช่น Access Token และ Username Password เป็นต้น ต้องไม่ประกอบด้วยข้อมูล Sensitive หรือข้อมูลที่บุคคลอื่นสามารถเข้าถึงได้ การรับส่งข้อมูลที่ใช้ยืนยันตัวตนต้องทำผ่านช่องทางที่มีความมั่นคงปลอดภัย เช่น TLS เป็นต้น ทั้งนี้ต้องมีกลไกในการตรวจสอบว่าข้อมูลหรือปัจจัยที่ใช้ในการยืนยันตัวตนดังกล่าวไม่ถูกเพิกถอนหรือหมดอายุแล้ว นอกจากนี้ผู้ให้บริการ API ต้องมีการบริหารจัดการ Session ที่เหมาะสม มีระยะเวลาสั้นเท่าที่จำเป็นต่อการทำธุรกรรม และตรวจสอบเป็นประจำเพื่อยุติ Session ที่ไม่ถูกใช้งานแล้วที่คงค้างอยู่ในระบบ

1.2. [กระบวนการยืนยันตัวตนทางด้านผู้ให้บริการ หรือ Server authentication] กระบวนการที่ผู้ใช้บริการตรวจสอบความถูกต้องของผู้ให้บริการว่าเป็นผู้ให้บริการตัวจริง สำหรับการขอใช้

บริการต่าง ๆ โดยแนวทางที่พัฒนาใช้กันส่วนใหญ่คือการตรวจสอบใบรับรองอิเล็กทรอนิกส์ (Certificate) ของผู้ให้บริการ ซึ่งต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

- การตรวจสอบว่าใบรับรองอิเล็กทรอนิกส์ถูกออกโดยผู้ให้บริการ CA (Certification Authority) ที่น่าเชื่อถือ
- การตรวจสอบว่าใบรับรองอิเล็กทรอนิกส์ไม่หมดอายุ
- การตรวจสอบว่าใบรับรองอิเล็กทรอนิกส์ไม่ถูกเพิกถอน
- การตรวจสอบว่าใบรับรองอิเล็กทรอนิกส์มีชื่อตรงกับชื่อโดเมนของบริการ

ตัวอย่างของมาตรฐานสากลหรือแนวปฏิบัติที่ดีด้านความมั่นคงปลอดภัยที่อาจใช้อ้างอิง เช่น มาตรฐาน OpenID Connect หรือ SAML (Security Assertion Markup Language) เป็นต้น

(2) [กระบวนการตรวจสอบสิทธิ์การใช้งาน (Authorization)] คือกระบวนการตรวจสอบเพื่อกำหนดทรัพยากรที่ผู้ใช้บริการสามารถเข้าถึงได้ ผู้ให้บริการ API ต้องมีกลไกในการจัดการและให้สิทธิ์ และตรวจสอบสิทธิ์ก่อนเสมอ รวมทั้งควบคุมการเข้าถึงของผู้ใช้บริการ API อย่างเข้มงวด ให้สามารถเข้าถึงได้เฉพาะข้อมูลที่ตนเองมีสิทธิ์เท่านั้น หรือเฉพาะฟังก์ชัน API ที่กำหนดไว้เท่านั้น เช่น API ฝั่งผู้ใช้บริการที่มีสิทธิ์แบบอ่านอย่างเดียวไม่ควรได้รับอนุญาตให้เข้าถึงปลายทางที่มีฟังก์ชันการทำงานของ API ฝั่งผู้ดูแลระบบ

นอกจากนี้ ผู้ให้บริการ API ต้องกำหนดกระบวนการให้มีการยืนยันตัวตนและการให้สิทธิ์แก่ผู้ใช้บริการ API ทุกครั้งก่อนการเข้าถึงข้อมูลและบริการผ่าน API โดยต้องมีความเหมาะสมกับลักษณะของผู้ใช้บริการ API ที่แตกต่างกัน เช่น เป็นระบบคอมพิวเตอร์ เป็นนิติบุคคล หรือเป็นบุคคลธรรมดา ตัวอย่างของมาตรฐานสากลหรือแนวปฏิบัติที่ดีด้านความมั่นคงปลอดภัยที่อาจใช้อ้างอิง เช่น มาตรฐาน OAuth 2.0 เป็นต้น หากมีการใช้งานการยืนยันตัวตนและการให้สิทธิ์โดยใช้ Token-based ควรคำนึงถึงความมั่นคงปลอดภัยของ Token และภัยจากการโจมตีซ้ำ (Replay Attack) รวมทั้ง ควรหาแนวการป้องกันภัยดังกล่าวตามมาตรฐานของ token ที่ใช้งาน เช่น การเลือกใช้ Strong encryption algorithm สำหรับการสร้าง Token การกำหนดอายุการใช้งานของ Token แบบจำกัด (Short-lived token) หรือการใช้งานค่า nonce เป็นต้น

(3) [การรักษาความลับของข้อมูลและการตรวจสอบความถูกต้องของข้อมูล (Data Confidentiality and Integrity)] ผู้ให้บริการ API ต้องป้องกันไม่ให้เปิดเผยข้อมูลเกินความจำเป็น (Excessive Data Exposure) การควบคุมข้อมูลที่เกิดจากการประมวลผล API ไม่ส่งข้อมูลมากเกินกว่าความจำเป็นต่อการเรียกใช้งาน รวมทั้ง ป้องกันการฝัง Sensitive Information โดยเฉพาะอย่างยิ่งข้อมูลสำคัญ เช่น

รหัสผ่าน กุญแจเข้ารหัส เป็นต้น ซึ่งข้อมูลสำคัญดังกล่าวอาจถูกฝังอยู่ใน Source code หรือ API Specification โดยแนวทางการรักษาความมั่นคงปลอดภัยของข้อมูลต้องประกอบไปด้วยอย่างน้อยดังต่อไปนี้

3.1. [การรักษาความลับของข้อมูล หรือ Data Confidentiality] คือการเก็บข้อมูลเป็นความลับและอนุญาตให้เฉพาะผู้ได้รับอนุญาตสามารถอ่านและเข้าถึงข้อมูลได้เท่านั้น กล่าวคือหากข้อความที่รับส่งระหว่าง API ผู้ใช้บริการและผู้ให้บริการนั้นประกอบด้วยข้อมูลที่มีความอ่อนไหว มีความสำคัญ หรือ Sensitive Information ข้อมูลในส่วนนี้ควรจะต้องถูกเก็บเป็นความลับโดยใช้การเข้ารหัสที่รัดกุม ใช้ความยาวกุญแจที่เหมาะสม สอดคล้องกับนโยบายหรือมาตรฐานการบริหารจัดการกุญแจเข้ารหัส และมีการควบคุมสิทธิ์การเข้าถึงอย่างรัดกุม เพื่อป้องกันการเข้าถึงจากบุคคลที่ไม่ได้รับอนุญาต

3.2. [การตรวจสอบความถูกต้องของข้อมูล หรือ Data Integrity] คือการตรวจสอบข้อมูลที่รับส่งระหว่าง API ผู้ใช้บริการและผู้ให้บริการว่าข้อมูลมีความถูกต้องและสมบูรณ์ครบถ้วน ไม่ถูกเปลี่ยนแปลงแก้ไขจากบุคคลภายนอกหรือบุคคลที่ไม่ได้รับอนุญาต

ตัวอย่างแนวปฏิบัติสำหรับการรักษาความลับของข้อมูลและการตรวจสอบความถูกต้องของข้อมูล คือ การรับส่งข้อมูลผ่านช่องทางที่เชื่อมต่อระบบด้วยโปรโตคอลที่เข้ารหัสเท่านั้น (HTTPS) เพื่อป้องกันการถูกดักจับและแก้ไขข้อมูลสำคัญระหว่างทางและต้องอนุญาตการเชื่อมต่อเฉพาะเทคโนโลยีที่มีความมั่นคงปลอดภัยตามมาตรฐานสากล เช่น TLS 1.2 สำหรับ Transport layer security และสำหรับการรับส่งข้อมูลที่มีความสำคัญสูงหรือ High Risk Data ควรพิจารณาถึงแนวทางการป้องกันภัยการเข้ารหัสในระดับแอปพลิเคชันเพื่อยกระดับการป้องกันข้อมูลที่มีความสำคัญสูงรั่วไหล เช่น ตามมาตรฐาน JWT (JSON Web Token), JWS (JSON Web Signature), JWE (JSON Web Encryption) และ WS-Security (Web Services Security) สำหรับ Message level integrity and encryption เป็นต้น

(4) [การรักษาความมั่นคงปลอดภัยด้านการสื่อสาร (Secure Communication)] คือ การสื่อสารระหว่าง API ผู้ใช้บริการและผู้ให้บริการควรเกิดขึ้นหลังจากการตรวจสอบความถูกต้องร่วมกันและผ่านช่องทางการเข้ารหัสที่เป็นมาตรฐาน เช่น การใช้โปรโตคอล TLS ร่วมกัน การติดตั้งใบรับรองอิเล็กทรอนิกส์ (Certificate) ที่ถูกต้องและเป็นปัจจุบันตามนโยบายความมั่นคงปลอดภัยของธนาคารบนเซิร์ฟเวอร์ API หรือ API Gateway รวมถึงการบังคับใช้ TLS และอนุญาตเฉพาะชุดการเข้ารหัสที่มีความแข็งแรง Strong Cipher Suites รวมถึงการยกเลิก หรือเลิกใช้ชุดเข้ารหัสที่ไม่ปลอดภัย

(5) [การพัฒนาโปรแกรมและการกำหนดค่าที่ปลอดภัย (Secure Coding and Configuration)] ผู้ให้บริการ API ต้องมีแนวทางการพัฒนาโปรแกรมและแนวทางการตั้งค่าความมั่นคงปลอดภัยที่เกี่ยวข้องกับการใช้งาน API โดยยึดหลักตาม least functionality ซึ่งเป็นการกำหนดให้มีการใช้งานเฉพาะเท่าที่จำเป็น ครอบคลุมหลักการดังต่อไปนี้

5.1. [Data Validation] ผู้ให้บริการ API ต้องตรวจสอบข้อมูลทั้งกรณีที่รับมาจากผู้ใช้บริการ (Request) และกรณีที่ส่งออกไปให้ผู้ใช้บริการ (Response) เพื่อให้มั่นใจว่ามีความถูกต้องและความครบถ้วนทั้งข้อมูลและพารามิเตอร์ที่ใช้ประมวลผล ทำให้โปรแกรมสามารถประมวลผลได้ตรงตามที่คาดหวัง สอดคล้องกับข้อกำหนดของรูปแบบข้อมูล (Data format validation) และข้อกำหนดของพารามิเตอร์ตามที่กำหนดไว้ เช่น การตรวจสอบประเภทของข้อมูลว่าตรงตามข้อกำหนดของรูปแบบที่ต้องการหรือไม่ ซึ่งวิธีการเหล่านี้จะช่วยคัดกรองข้อมูลที่ผิดปกติหรือข้อมูลที่ไม่เข้ากับการทำงานของระบบออก เพื่อลดความเสี่ยงที่ระบบอาจจะเกิดข้อผิดพลาดต่าง ๆ รวมถึงมีแนวทางสำหรับป้องกันการถูกโจมตีประเภท Injection ในรูปแบบต่าง ๆ เช่น SQL Injection, OS Command Injection เป็นต้น

5.2. [Security Header] ผู้ให้บริการ API ต้องมีแนวทางการตั้งค่าความมั่นคงปลอดภัยต่าง ๆ ของรายการเรียกใช้และรายการตอบสนอง (Request and Response Headers) ที่เกี่ยวข้องกับการใช้งาน API ให้เหมาะสม เช่น

- Cross-Origin-Resource-Sharing คือ กลไกการจำกัดการเข้าถึงข้อมูลจากการร้องของโดเมนอื่นนอกเหนือจากโดเมนที่ให้บริการ
- HTTP verbs คือ การกำหนด HTTP Request Method ที่อนุญาตให้ดำเนินการได้ และปิด HTTP Request Method อื่นที่ไม่จำเป็น
- การกำหนด Content-Type คือ การตรวจสอบ Content Type ของคำสั่งการเรียกใช้งานต้องระบุอย่างถูกต้อง

5.3. [Error Message] ผู้ให้บริการ API ต้องมีกลไกในการจัดการความผิดพลาดของระบบ โดยต้องจัดการการแสดงผลของ Error ทั้งหมดที่อาจเกิดขึ้น โดยต้องไม่แสดงข้อมูลรายละเอียดของ Error มากเกินความจำเป็น เนื่องจากข้อมูลบางอย่างอาจส่งผลกระทบต่อความมั่นคงปลอดภัยของระบบ

(6) [การจัดเก็บข้อมูลบันทึกเหตุการณ์ และการเฝ้าระวัง (Audit Log and Monitoring)] ผู้ให้บริการ API ต้องมีการจัดเก็บข้อมูลบันทึกเหตุการณ์ (Audit log) ของการใช้งาน API โดยมีรายละเอียดที่ครบถ้วนเพียงพอที่ใช้เป็นหลักฐานในการตรวจสอบที่สามารถระบุตัวบุคคลผู้กระทำได้ เช่น มีรายละเอียด

ระบุว่า ใครทำอะไร ที่ไหน และเมื่อไหร่ โดยทั่วไปแล้วตามกฎหมายทางคอมพิวเตอร์ระบุว่าจะต้องเก็บรักษาข้อมูลเหล่านั้นไว้ไม่ต่ำกว่า 90 วัน หรืออาจจะมากกว่าขึ้นกับนโยบายของทางธนาคาร และบันทึกเวลาอิเล็กทรอนิกส์ (timestamp) ได้อย่างถูกต้อง รวมทั้งมีมาตรการควบคุมไม่ให้มีการเปลี่ยนแปลงแก้ไข Audit log โดยไม่ได้รับอนุญาต ไม่อนุญาตให้ผู้ดูแลระบบสามารถเข้าไปปรับเปลี่ยนข้อมูล Audit log ที่เก็บรักษาไว้ได้ ต้องมั่นใจว่าการจัดเก็บข้อมูลบันทึกเหตุการณ์จะไม่ถูกแก้ไข และสามารถระบุตัวตนได้ว่าบุคคลใดมีสิทธิ์ในการเข้าถึง Audit log ได้บ้าง นอกจากนี้สิ่งสำคัญที่ต้องระวังคือ หากมีการจัดเก็บข้อมูลบันทึกเหตุการณ์ที่ประกอบด้วยข้อมูลอ่อนไหว หรือ Sensitive data หรือข้อมูลที่อยู่ในชั้นความลับ อยู่ในชุดข้อมูลที่ถูกจัดเก็บ ต้องปฏิบัติให้สอดคล้องกับแนวทางการรักษาความมั่นคงปลอดภัยตามระดับชั้นความลับข้อมูล

สำหรับกระบวนการเฝ้าระวังควรมีมาตรการ เพื่อตรวจจับ แจ้งเตือน และตอบสนองต่อพฤติกรรมที่ผิดปกติ เช่น การพยายามยืนยันตัวตนที่ล้มเหลว การปฏิเสธการเข้าถึง ข้อผิดพลาดในการตรวจสอบอินพุต และข้อผิดพลาดของพารามิเตอร์ การเรียกใช้ Token ซ้ำ ๆ พฤติกรรมการโจมตีเซสชัน รวมทั้งการเกิดเหตุขัดข้องต่าง ๆ เป็นต้น และกระบวนการเฝ้าระวังกรณีที่มีปริมาณการเรียกใช้งานจำนวนมากที่อาจส่งผลกระทบต่อการให้บริการได้ เช่น การเฝ้าระวังในช่วงสิ้นเดือนที่อาจมีปริมาณการใช้งานเป็นจำนวนมาก เป็นต้น

(7) [ความเพียงพอของทรัพยากร (Resource Sufficiency)] ผู้ให้บริการ API ต้องมีการจัดการทรัพยากรอย่างมีประสิทธิภาพ มีมาตรการตรวจสอบให้แน่ใจว่า API ได้รับการปกป้องจากการใช้งานทรัพยากรที่มากเกินไป และบริการ API จะไม่หยุดชะงัก โดยแนวทางการจัดการทรัพยากรให้เกิดความเพียงพอต้องดำเนินการอย่างน้อยในเรื่องดังต่อไปนี้

7.1 [Rate Limit & Circuit Breaker] ผู้ให้บริการและใช้บริการ API ต้องมีแนวทางควบคุมการใช้งานทรัพยากรบนเซิร์ฟเวอร์อย่างมีประสิทธิภาพ เช่น การจำกัดจำนวนข้อความ API ต่อวินาที การกำหนดขนาดของข้อความ API ที่สามารถรับและนำไปประมวลผลทำงานต่อได้ เพื่อป้องกันไม่ให้เกิดการใช้งานทรัพยากรของเซิร์ฟเวอร์ที่เกินความสามารถที่ระบบจะรองรับได้ และป้องกันการโจมตีที่ทำให้เซิร์ฟเวอร์ไม่สามารถให้บริการได้ (DoS) เช่น การกำหนดปริมาณการเรียกใช้งานรายบุคคลหรือรายบริการ การกำหนดขนาดของข้อมูลที่รับส่ง เป็นต้น รวมทั้งควรพิจารณาแนวทางในการตัดการเชื่อมต่อ สำหรับกรณีที่ระบบมีการเรียกใช้งาน API จากภายนอก หรือมาตรการอื่น ๆ ที่สามารถป้องกันได้ เพื่อเป็นการป้องกันผลกระทบของระบบอันเกิดจากการที่ API ภายนอกดังกล่าวตอบสนองช้า หรือไม่ตอบสนอง รวมทั้ง แนวทางการกลับมาเปิดให้บริการเมื่อสถานการณ์กลับมาสู่ภาวะปกติ โดยคำนึงถึงความมั่นคงปลอดภัยความสอดคล้องกับธุรกิจและเหมาะสมกับการใช้งาน

7.2 [DDoS Protection] ผู้ให้บริการ API ต้องมีแนวทางเชิงป้องกันภัย DDoS Attack ในระดับเครือข่าย หรือควรพิจารณาเพิ่มเติม DDoS Protection ในระดับแอปพลิเคชันสำหรับการเปิดให้บริการ API ที่มีความสำคัญ มีลักษณะความต้องการความพร้อมใช้งาน (Availability) เช่น การใช้ WAF หรือมาตรการอื่น ๆ ที่สามารถป้องกันได้ เพื่อป้องกันระบบถูกโจมตีจนไม่สามารถให้บริการได้ เช่น HTTPs Flood, Application-Level Exploitation เป็นต้น

7.3 [Request and Response size] ผู้ให้บริการ API ต้องมีแนวทางจำกัดขนาดของข้อความในการร้องขอบริการ (Request payload size) โดยควรกำหนดค่าให้เหมาะสมกับรูปแบบการให้บริการ เพื่อป้องกันความเสี่ยงระบบหยุดชะงักจากการใช้งานทรัพยากรที่มากเกินไป รวมทั้งต้องมีแนวทางจำกัดปริมาณข้อมูลที่จะส่งกลับไปยังผู้ใช้บริการ (Number of records per page to return in a single request response) เช่น การทำ Pagination เพื่อจำกัดจำนวนแถวข้อมูลต่อหน้าที่จะส่งกลับ เป็นต้น

ทั้งนี้ ควรพิจารณาเลือกใช้โดยระมัดระวัง ไม่นำมาตรฐานหรือแนวปฏิบัติที่ล้าสมัยหรือเคยได้รับการพิสูจน์แล้วว่ามีช่องโหว่ด้านความมั่นคงปลอดภัยมาใช้งาน รวมทั้ง อาจพิจารณาเพิ่มเติมมาตรฐานการควบคุมอื่นให้สอดคล้องกับมาตรฐานสากลอื่น ๆ ที่เกี่ยวข้องได้ เช่น แนวทางการควบคุมจากมาตรฐาน PCI-DSS เป็นต้น

นอกจากนี้ เพื่อให้แนวทางปฏิบัติที่กล่าวมาข้างต้นทั้ง 7 หัวข้อ สามารถนำไปพิจารณาปรับใช้ได้อย่างเหมาะสมสอดคล้องกับการพัฒนา API เพื่อให้บริการขององค์กร เพื่อสร้างความมั่นคงปลอดภัยตามมาตรฐานสากลและสอดคล้องกับระดับความเสี่ยงของประเภท API และข้อมูล โดยรายละเอียดเกี่ยวกับการประเมินระดับความเสี่ยงตั้งต้นของ API และการกำหนดมาตรฐานการควบคุมด้านความมั่นคงปลอดภัยสำหรับการใช้เทคโนโลยี API เพื่อให้บริการทางการเงิน อธิบายในภาคผนวก 2

หลักการที่ 4 ความสัมพันธ์ระหว่างผู้ให้บริการ API และผู้ใช้บริการ API (Contractual Relationship)

ผลลัพธ์ที่คาดหวัง : ความสัมพันธ์ระหว่างผู้ให้บริการ API และผู้ใช้บริการ API พึงได้รับการบริหารจัดการอย่างเหมาะสมตามความสัมพันธ์ทางธุรกิจภายใต้ข้อสัญญาที่มีอยู่ระหว่างกัน กฎหมาย และหลักเกณฑ์การกำกับดูแลที่เกี่ยวข้องสอดคล้องกับระดับความเสี่ยง รวมทั้งมีการกำหนดบทบาทหน้าที่ความรับผิดชอบระหว่างกันที่ชัดเจน และมีทบทวนมาตรฐานความมั่นคงปลอดภัยระหว่างกันอย่างสม่ำเสมอ

การปฏิบัติตามหลักการนี้จะช่วยให้มีการบริหารจัดการความสัมพันธ์ระหว่างผู้ให้บริการและผู้ใช้บริการ API อย่างเหมาะสมตามวัตถุประสงค์ข้อที่ 1 ของแนวปฏิบัติฉบับนี้ และช่วยปิดความเสี่ยงในด้าน

การเชื่อมต่อกับหน่วยงานภายนอกที่มีมาตรการรักษาความมั่นคงปลอดภัยที่ไม่เพียงพอ (Inadequate Security Measures of Third Party)

แนวทางที่พึงปฏิบัติ

(1) **[การคัดเลือกผู้ให้บริการ]** ผู้ใช้บริการ API ต้องมีกระบวนการในการประเมินและคัดเลือกผู้ให้บริการ API ที่จะทำการเชื่อมต่อ โดยครอบคลุมถึงการประเมินความเสี่ยงจากการเชื่อมต่อกับหน่วยงานภายนอก ให้สอดคล้องกับแนวทางประเมินความเสี่ยงที่เกี่ยวข้องของการใช้บริการ การเชื่อมต่อหรือการเข้าถึงข้อมูลจากบุคคลภายนอก เช่น ฐานะทางการเงิน ชื่อเสียง ศักยภาพการดำเนินธุรกิจและศักยภาพด้านเทคโนโลยีสารสนเทศ การบริหารจัดการภายในองค์กร การบริหารความเสี่ยง การดูแลรักษาความมั่นคงปลอดภัยด้านเทคโนโลยีสารสนเทศและข้อมูล กระบวนการรับมือและแก้ไขปัญหา การคุ้มครองผู้ใช้บริการ และการคุ้มครองข้อมูลส่วนบุคคล เป็นต้น

(2) **[การลงทะเบียนและคัดเลือกผู้ใช้บริการ]** ผู้ให้บริการ API ต้องมีกระบวนการในการรับลงทะเบียนและการคัดเลือกผู้ใช้บริการ API โดยครอบคลุมถึงการประเมินความเสี่ยงจากการเชื่อมต่อกับหน่วยงานภายนอก เช่น การรู้จักตัวตนและการทราบข้อเท็จจริงเกี่ยวกับตัวตน การติดตามควบคุม การกำหนดหน้าที่ความรับผิด การสร้างและส่งมอบข้อมูลหรือปัจจัยที่ใช้ในการยืนยันตัวตน อย่างเหมาะสม ง่าย รวดเร็ว และสอดคล้องกับระดับความเสี่ยงและประเภท API โดยต้องคำนึงถึงปัจจัยต่าง ๆ เช่น ฐานะทางการเงิน ชื่อเสียง ศักยภาพการดำเนินธุรกิจและศักยภาพด้านเทคโนโลยีสารสนเทศ การบริหารจัดการภายในองค์กร การบริหารความเสี่ยง การดูแลรักษาความมั่นคงปลอดภัยด้านเทคโนโลยีสารสนเทศและข้อมูล กระบวนการรับมือและแก้ไขปัญหา การคุ้มครองผู้ใช้บริการ และการคุ้มครองข้อมูลส่วนบุคคล เป็นต้น

การพิจารณาคัดเลือกผู้ใช้บริการ API ผู้ให้บริการ API ต้องไม่มีข้อกำหนดที่เป็นการเลือกปฏิบัติ หรือการปิดกั้นไม่ให้สามารถเข้าใช้บริการได้อย่างไม่เป็นธรรม

ทั้งนี้ สำหรับการใช้บริการ API ข้อมูลที่เปิดเผยต่อสาธารณะ ผู้ให้บริการ API อาจไม่จำเป็นต้องกำหนดให้ผู้ใช้บริการ API ต้องลงทะเบียนก่อนใช้บริการก็ได้ ขึ้นอยู่กับการบริหารความเสี่ยงของผู้ให้บริการ API แต่ละราย

(3) **[เอกสารที่ใช้ประกอบการคัดเลือก]** ผู้ให้บริการ API ต้องขอข้อมูล เอกสาร และรายละเอียดต่าง ๆ ที่มีความถูกต้อง ครบถ้วน จากผู้ใช้บริการ API เพื่อให้กระบวนการการลงทะเบียน การพิสูจน์ตัวตน และการตรวจสอบข้อเท็จจริง เป็นไปอย่างเหมาะสมและเชื่อถือได้ และต้องแจ้งให้ผู้ใช้บริการ API ปรับปรุงข้อมูลให้เป็นปัจจุบันหากมีการเปลี่ยนแปลงในเอกสารที่ได้เคยส่งมอบไว้

ข้อมูลที่ขอจากผู้ใช้บริการ API ควรครอบคลุมถึงประเด็นด้านต่าง ๆ ตามระดับความเสี่ยงและประเภทของข้อมูลหรือบริการที่ให้บริการ เช่น แนวทางการกำกับดูแลกิจการและการบริหารความเสี่ยง แนว

ทางการบริหารความเสี่ยงด้านเทคโนโลยีสารสนเทศ แนวทางการคุ้มครองข้อมูลส่วนบุคคล แนวทางการคุ้มครองผู้ใช้บริการ แนวทางการให้บริการอย่างต่อเนื่องในสภาวะฉุกเฉิน และ แนวทางการบริหารจัดการผู้ให้บริการภายนอก

(4) **[ข้อสัญญาระหว่างกัน]** ผู้ให้บริการ API และ ผู้ใช้บริการ API ต้องปฏิบัติตามข้อสัญญา ข้อตกลงทางธุรกิจ กฎหมายและหลักเกณฑ์ที่เกี่ยวข้องกับความสัมพันธ์ระหว่างคู่สัญญา

ข้อสัญญาระหว่างผู้ให้บริการ API และผู้ใช้บริการ API ต้องทำเป็นลายลักษณะอักษรหรือรูปแบบทางอิเล็กทรอนิกส์และครอบคลุมในประเด็นต่าง ๆ โดยเหมาะสมตามความสัมพันธ์ระหว่างกันและระดับความเสี่ยงของธุรกรรม ทั้งนี้ข้อสัญญาดังกล่าวต้องไม่มีข้อสัญญาที่เป็นการสร้างอุปสรรคขัดขวางการเข้าถึงบริการของผู้ใช้บริการ API หรือเป็นการปิดกั้นไม่ให้ผู้ใช้บริการ API ไปใช้บริการจากผู้ให้บริการ API รายอื่นโดยไม่เป็นธรรม

(5) **[การทบทวนมาตรฐานความมั่นคงปลอดภัย 3rd Party]** ผู้ให้บริการ API และผู้ใช้บริการ API ต้องมีการทบทวนมาตรฐานในการรักษาความมั่นคงปลอดภัยของระหว่างผู้ให้บริการและผู้ใช้บริการ API ให้ยังคงเป็นไปตามที่เคยตกลงกันอย่างสม่ำเสมอตามความเหมาะสม ทั้งนี้ในการทบทวน ผู้ให้บริการ API ควรพิจารณาทบทวนด้วยว่ามาตรฐานที่กำหนดไว้นั้นยังคงมีความมั่นคงปลอดภัยตามมาตรฐานสากลที่ใช้อยู่อย่างแพร่หลายในปัจจุบัน

(6) **[Dispute Resolution between API Provider and API Consumer และความรับผิดชอบต่อความเสียหาย]** ผู้ให้บริการ API และผู้ใช้บริการ API ต้องมีการกำหนดบทบาทหน้าที่และความรับผิดชอบต่อความเสียหาย รวมทั้งตกลงร่วมกันถึงกระบวนการแก้ไขปัญหาที่อาจเกิดขึ้นระหว่างกัน หรือกรณีที่เกิดการละเมิดสัญญาหรือข้อตกลงระหว่างกัน เช่น หน้าที่และความรับผิดของแต่ละฝ่าย กระบวนการสื่อสารที่ชัดเจน เป็นต้น โดยต้องสอดคล้องกับลักษณะความสัมพันธ์ในทางธุรกิจระหว่างกัน ซึ่งปัญหาข้อขัดแย้งดังกล่าวอาจเกิดจากการใช้บริการของผู้ใช้บริการทางการเงินหรือผู้ใช้บริการ API หรือเกิดจากการให้บริการของผู้ให้บริการ API เอง เช่น ความเสียหายที่เกิดจากการทำธุรกรรมโดยไม่ได้รับอนุญาต ทำธุรกรรมไม่สำเร็จ เป็นต้น

(7) **[การยกเลิกและการสิ้นสุดสัญญาหรือข้อตกลง]** สำหรับกรณีที่เป็นผู้ใช้บริการ API ต้องมีมาตรฐานหรือระเบียบปฏิบัติว่าด้วยการยกเลิกและสิ้นสุดสัญญาหรือข้อตกลง โดยคำนึงถึงความต่อเนื่องในการให้บริการ และความมั่นคงปลอดภัยด้านเทคโนโลยีสารสนเทศ เช่น การทำลายข้อมูล หลังการใช้งาน หรือการยกเลิกการใช้งาน เป็นต้น

(8) **[การทดสอบร่วมกัน]** ผู้ให้บริการ API ควรสนับสนุนให้ผู้ใช้บริการ API ทดสอบการใช้บริการ และผู้ใช้บริการ API ควรทดสอบการใช้บริการให้มั่นใจว่าระบบสามารถทำงานได้อย่างถูกต้องก่อนใช้บริการจริงหากมีช่องทางที่สามารถทำการทดสอบได้

หลักการที่ 5 การเปิดเผยข้อมูลการให้บริการ API (API Service Information Disclosure)

ผลลัพธ์ที่คาดหวัง : ข้อมูลพื้นฐานที่จำเป็นต่อการใช้งาน API พึงได้รับการเปิดเผยอย่างเพียงพอต่อการตัดสินใจเลือกใช้งาน หรือนำ API ไปใช้งานได้อย่างมีประสิทธิภาพ มีช่องทางในการทดสอบให้แก่ผู้ใช้งาน และส่งเสริมให้ผู้ใช้บริการมีสภาวะแวดล้อมที่เอื้ออำนวยให้สามารถนำ API ไปใช้พัฒนานวัตกรรมได้อย่างรวดเร็วยิ่งขึ้น

การปฏิบัติตามหลักการนี้จะช่วยให้มีสภาพแวดล้อมที่สนับสนุนการต่อยอดนวัตกรรมทางการเงิน ซึ่งจะช่วยลดระยะเวลาและอุปสรรคของนักพัฒนาที่นำ API ไปประยุกต์ใช้ ตามวัตถุประสงค์ข้อที่ 2 ของแนวปฏิบัติฉบับนี้

แนวทางที่พึงปฏิบัติ

(1) **[ช่องทางการเผยแพร่ API Specification]** ในกรณีที่เป็นการให้บริการ API แก่ผู้ใช้บริการ API ที่เป็นหน่วยงานภายนอก ผู้ให้บริการ API ต้องจัดให้มีเครื่องมือที่จำเป็นแก่ผู้ใช้บริการ API ในการนำ API ไปใช้งาน เช่น API Developer Portal หรือวิธีการอื่น ๆ ในลักษณะคล้ายกัน เพื่อเป็นช่องทางให้ผู้ใช้บริการ API สามารถศึกษารายละเอียดเกี่ยวกับการพัฒนาและอาจสามารถอำนวยความสะดวกในด้านอื่น ๆ เช่น API Specification การลงทะเบียนเข้าใช้บริการ การทดสอบการใช้งาน API การบริหารจัดการข้อมูลหรือปัจจัยที่ใช้ในการยืนยันตัวตน และการติดต่อขอรับการสนับสนุนจากผู้ให้บริการ API เป็นต้น เพื่อแก้ไขปัญหาที่พบได้ด้วยตนเอง

ทั้งนี้ เครื่องมือดังกล่าวควรมีรูปแบบที่สอดคล้องเหมาะสมกับลักษณะการให้บริการ API และจำนวนผู้ใช้บริการ API โดยควรคำนึงถึงประสบการใช้งานของผู้ใช้บริการ API เพื่อให้สามารถนำไปใช้พัฒนาได้โดยง่าย สะดวก และรวดเร็ว

(2) **[ข้อมูลที่ต้องเปิดเผย]** ผู้ให้บริการ API ต้องเผยแพร่ข้อมูลรายละเอียดและเอกสารต่าง ๆ ที่เกี่ยวข้องและจำเป็นต่อการนำไปใช้งาน และต่อการยกเลิกการใช้งานของผู้ใช้บริการ API ด้วยภาษาที่สั้นกระชับ ได้ใจความ และมีตัวอย่างประกอบที่ชัดเจน เช่น

1) ข้อตกลงทางธุรกิจและเงื่อนไขในการใช้บริการ
2) วิธีการลงทะเบียนเพื่อใช้บริการ
3) API Specification ซึ่งควรมีตัวอย่าง Request Response ตัวอย่างการเขียนโปรแกรมเพื่อเรียกใช้งาน API และขั้นตอนการเรียกใช้งานเพื่อให้เข้าใจกลไกและลำดับการทำงานของ API
4) วิธีการยืนยันตัวตน

5) ช่องทางการเชื่อมต่อและรูปแบบของเครือข่ายคอมพิวเตอร์ที่ใช้
6) สถานะการให้บริการ API ตาม API lifecycle
7) รายละเอียดเกี่ยวกับค่าธรรมเนียมการใช้บริการและ Subscription Plan แต่ละประเภท
8) ข้อมูลติดต่อเพื่อขอรับการสนับสนุนการให้บริการหรือแจ้งปัญหาการใช้งาน
9) วิธีการยกเลิกการใช้งาน หรือการหยุดใช้งานชั่วคราว

(3) **[Sandbox Environment]** ผู้ให้บริการ API ควรมีช่องทางให้ผู้ใช้งาน API สามารถนำเอาระบบที่ตนได้พัฒนามาทดสอบกับ API ชุดทดสอบ โดย API ชุดทดสอบควรมีระบบทดสอบและข้อมูลชุดทดสอบเพื่อให้การทดสอบมีความใกล้เคียงกับการใช้งานจริงด้วย ทั้งนี้ ระบบและข้อมูลที่นำมาใช้กับ API ชุดทดสอบต้องไม่เป็นระบบและข้อมูลของผู้ใช้บริการจริง

หลักการที่ 6 การคุ้มครองผู้ใช้บริการทางการเงิน (Customer Protection)

ผลลัพธ์ที่คาดหวัง : ผู้ใช้บริการทางการเงินพึงได้รับการคุ้มครองตามกฎหมายและตามสิทธิต่าง ๆ ที่ตนมี ได้รับข้อมูลที่เพียงพอ มีช่องทางในการแจ้งปัญหาและข้อร้องเรียน ได้รับการดูแล และการเยียวยาอย่างเหมาะสมเมื่อได้รับความเสียหายจากการใช้บริการ

การปฏิบัติตามหลักการนี้จะช่วยให้ผู้ใช้บริการทางการเงินได้รับการคุ้มครองในด้านต่าง ๆ ตามวัตถุประสงค์ข้อที่ 3 ของแนวปฏิบัติฉบับนี้

แนวทางที่พึงปฏิบัติ

(1) **[ต้องปฏิบัติให้สอดคล้องกับ PDPA]** ผู้ให้บริการ API และ/หรือผู้ใช้บริการ API แล้วแต่กรณีต้องปฏิบัติตามกฎหมายว่าด้วยการคุ้มครองข้อมูลส่วนบุคคล โดยเฉพาะอย่างยิ่ง ในเรื่องการขอและถอนความยินยอม การแจ้งวัตถุประสงค์ในการเก็บรวบรวม ใช้ และเปิดเผยข้อมูล การใช้และเปิดเผยข้อมูลภายใต้วัตถุประสงค์ที่เคยแจ้งไว้ การแจ้งเหตุการณ์ละเมิดข้อมูล และสิทธิ์ของเจ้าของข้อมูล

ทั้งนี้ ในกรณีที่เป็นข้อมูลที่ไม่ใช่ข้อมูลส่วนบุคคลตามกฎหมายว่าด้วยการคุ้มครองข้อมูลส่วนบุคคล เช่น ข้อมูลบัญชีเงินฝากของนิติบุคคล ผู้ให้บริการ API และ/หรือผู้ใช้บริการ API แล้วแต่กรณี มีแนวทางบริหารจัดการความเสี่ยงที่เกี่ยวข้องกับข้อมูลนั้นตามข้อกำหนดที่ใช้กับข้อมูลส่วนบุคคลโดยอนุโลมหรือตามความเหมาะสมหรือ โดยคำนึงถึงลักษณะความเสี่ยงของข้อมูลหรือบริการนั้น

(2) **[ช่องทางแจ้งปัญหาและข้อร้องเรียน]** ผู้ให้บริการ API และ/หรือผู้ใช้บริการ API แล้วแต่กรณีต้องมีช่องทาง ระบบและกระบวนการในการรับแจ้งปัญหาหรือข้อร้องเรียนจากผู้ใช้บริการทางการเงิน และต้องแจ้งให้ผู้ใช้บริการทราบถึงวิธีการในการเข้าถึงหรือติดต่อเพื่อแจ้งปัญหาหรือร้องเรียน โดยต้องมีการบริหารจัดการ ติดตาม เยียวยา เพื่อการแก้ไขปัญหาที่รวดเร็ว เพียงพอ และเหมาะสม เพื่อให้การดำเนินการเป็นไป

ตามระยะเวลาที่กำหนด กำหนดผู้มีหน้าที่รับผิดชอบชัดเจน และให้ข้อมูลสถานะของการดำเนินการแก้ไขปัญหาหรือขอร้องเรียนให้ผู้ใช้บริการทราบเมื่อผู้ใช้บริการร้องขอ พร้อมทั้งมีกระบวนการในการป้องกันไม่ให้ปัญหาเกิดขึ้นอีก

(3) **[ให้ความรู้แก่พนักงาน]** ผู้ให้บริการ API และ/หรือผู้ใช้บริการ API แล้วแต่กรณี ต้องสื่อสารและให้ความรู้พร้อมทั้งประเมินผลเกี่ยวกับการให้บริการที่เกี่ยวข้องแก่พนักงานซึ่งเป็นผู้ให้การสนับสนุนหรือเกี่ยวข้องกับการให้บริการ เช่น พนักงาน Call Center หรือ พนักงาน IT Support ของหน่วยงานตน เพื่อให้พนักงานมีความรู้ความสามารถเพียงพอในการสนับสนุนให้ผู้ใช้บริการทางการเงินได้รับบริการที่ดี และได้รับข้อมูลที่ถูกต้อง

(4) **[ให้ความรู้แก่ผู้ใช้บริการ]** ผู้ให้บริการ API และ/หรือผู้ใช้บริการ API แล้วแต่กรณีต้องสื่อสารและให้ความรู้เกี่ยวกับการให้บริการที่เกี่ยวข้องแก่ผู้ใช้บริการทางการเงินอย่างเพียงพอต่อการตัดสินใจใช้บริการ เพื่อให้ผู้ใช้บริการใช้งานได้อย่างถูกต้อง เหมาะสม และปลอดภัย ซึ่งจะช่วยให้ผู้ใช้บริการได้รับประโยชน์จากการใช้บริการได้อย่างมีประสิทธิภาพและลดความเสี่ยงจากปัญหาที่อาจเกิดจากความเข้าใจที่ไม่ถูกต้อง

ภาคผนวก 1 ความรู้พื้นฐานเกี่ยวกับเทคโนโลยี API

1. ความหมายของ API

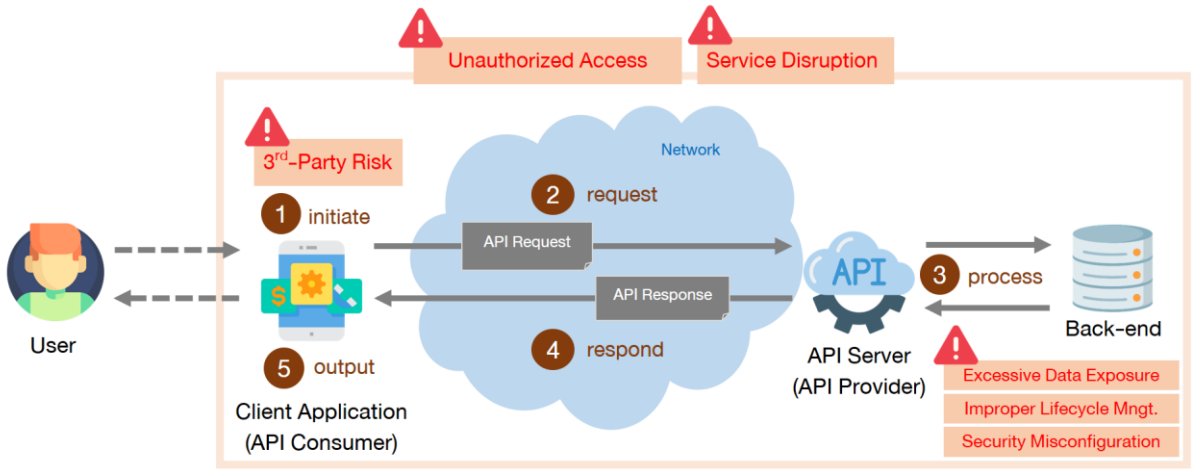
Application Programming Interface (API) หมายถึงชุดของคำสั่งหรือฟังก์ชันในโปรแกรมคอมพิวเตอร์ใด ๆ ซึ่งได้เตรียมไว้โดยมีข้อกำหนดกฎเกณฑ์จำเพาะ และได้เปิดเป็นช่องทางให้นักพัฒนาสามารถพัฒนาโปรแกรมคอมพิวเตอร์อื่นมาเรียกใช้งาน เพื่อสื่อสาร แลกเปลี่ยนข้อมูล หรือมีปฏิสัมพันธ์ต่าง ๆ ระหว่างกัน โดย API มีลักษณะที่สำคัญ เช่น

1. เป็นการสื่อสารระหว่างโปรแกรมคอมพิวเตอร์กับโปรแกรมคอมพิวเตอร์
2. เป็นการสื่อสารสองทาง ทั้งรับและส่ง
3. มีโครงสร้างของการรับส่งข้อมูลที่ชัดเจน
4. มีการเปิดเผยช่องทาง วิธีการ ข้อกำหนดกฎเกณฑ์ในการใช้งานให้ทราบล่วงหน้า
5. ไม่มีการเปิดเผยรายละเอียดขั้นตอนกระบวนการทำงานเบื้องหลัง

ทั้งนี้ เราสามารถพบเห็นการใช้งาน API ได้โดยทั่วไปในการทำงานร่วมกันระหว่างโปรแกรมคอมพิวเตอร์ต่าง ๆ เช่น ธนาคารใช้ API เป็นช่องทางสำหรับการสื่อสารข้อมูลระหว่าง Mobile Banking Application ที่อยู่บนโทรศัพท์มือถือของลูกค้ากับระบบคอมพิวเตอร์แม่ข่าย (Server) ที่อยู่ที่ศูนย์คอมพิวเตอร์ของธนาคาร หรือ บริษัทผู้ให้บริการ Ride Sharing ใช้บริการข้อมูลแผนที่จากผู้ให้บริการแผนที่เพื่อมาแสดงใน Mobile Application ของตนผ่าน API เป็นต้น ดังนั้น ด้วยความสามารถของเทคโนโลยี API ที่ทำให้การทำงานร่วมกันระหว่างโปรแกรมคอมพิวเตอร์สามารถเกิดขึ้นได้ เทคโนโลยี API จึงมีความสำคัญเป็นอย่างยิ่งต่อการพัฒนาโปรแกรมคอมพิวเตอร์ในปัจจุบัน

แม้ว่า API ที่ได้รับความนิยมเป็นอย่างมากในปัจจุบันคือ API ที่ทำงานอยู่บน HTTP Protocol ซึ่งมักถูกเรียกว่า Web API ซึ่งมีให้บริการในหลากหลายรูปแบบเช่น REST, SOAP, GraphQL และ gRPC เป็นต้น อย่างไรก็ตาม API ตามนิยามของแนวปฏิบัติฉบับนี้ เป็นนิยามตามความหมายอย่างกว้างของ API ที่ไม่ได้มีความเฉพาะเจาะจงกับเทคโนโลยีใดเทคโนโลยีหนึ่ง จึงอาจหมายถึงสิ่งอื่นที่ไม่ใช่ Web API ได้อีกด้วย

2. กระบวนการทำงานของ API ประกอบด้วย 5 ขั้นตอนหลักดังนี้



(1) เริ่มทำรายการ (Initiate) กระบวนการทำงานของ API โดยทั่วไปแล้ว จะเริ่มต้นจากการที่ Client Application ต้องการเรียกข้อมูลหรือใช้บริการจากผู้ให้บริการ API ซึ่งความต้องการดังกล่าวอาจขึ้นโดยการตัดสินใจโดยอัตโนมัติของ Client Application เอง หรืออาจเกิดจากการสั่งการขอผู้ใช้บริการก็ได้

ในมุมมองของผู้ให้บริการ API แล้ว หาก Client Application เป็นของหน่วยงานภายนอก ขั้นตอนนี้จะมีความเสี่ยงจากการเชื่อมต่อกับหน่วยงานภายนอกที่มีมาตรการรักษาความมั่นคงปลอดภัยที่ไม่เพียงพอ (Inadequate Security Measures of Third Party) ผู้ให้บริการ API จึงให้ความสำคัญในการระมัดระวังความเสี่ยงในด้านนี้ในขั้นตอนนี้ด้วย

(2) ส่งคำสั่งการทำรายการไปยัง API Provider (Request) Client Application สร้างชุดข้อมูล API Request เพื่อเรียกข้อมูลหรือใช้บริการจากผู้ให้บริการ API ตามข้อกำหนดกฎเกณฑ์ที่ผู้ให้บริการ API กำหนดไว้ และส่งผ่านเครือข่าย Network ไปยังผู้ให้บริการ API

(3) ประมวลผลรายการ (Process) เมื่อผู้ให้บริการ API ได้รับ API Request แล้ว ผู้ให้บริการ API จะดำเนินการประมวลผลโดยระบบต่าง ๆ ที่เกี่ยวข้อง เพื่อจัดเตรียมผลลัพธ์ข้อมูลหรือดำเนินการให้บริการตามที่ Client Application ร้องขอ

ขั้นตอนนี้มีความเสี่ยงหลายประการที่อาจเกิดขึ้นได้ เนื่องจากการบริหารจัดการ API หรือบริหารจัดการความเสี่ยงที่ไม่รัดกุมเพียงพอ เช่น การเปิดเผยข้อมูลเกินจำเป็น (Excessive Data Exposure) ความเสี่ยงที่อาจเกิดจากการบริหารจัดการวงจรชีวิตที่ไม่รัดกุม (Improper API Lifecycle Management) การตั้งค่าความมั่นคงปลอดภัยของระบบที่ผิดพลาด (Security Misconfiguration) ผู้ให้บริการ API จึงให้ความสำคัญในการระมัดระวังความเสี่ยงในด้านเหล่านี้ในขั้นตอนนี้ด้วย

(4) ส่งผลลัพธ์การทำรายการกลับไปยัง API Consumer (Respond) ผู้ให้บริการ API สร้างชุดข้อมูล API Response ที่ประกอบด้วยข้อมูลผลลัพธ์จากการประมวลผลในขั้นตอนก่อนหน้า เพื่อส่งผลลัพธ์ของข้อมูลที่ร้องขอหรือผลลัพธ์ของการให้บริการไปยัง Client Application

(5) นำผลลัพธ์ที่ได้มาแสดงหรือใช้งาน (Output) เมื่อ Client Application ได้รับ API Response แล้ว Client Application จะดำเนินการแสดงผลข้อมูลให้ผู้ใช้บริการทราบ หรือใช้งานผลลัพธ์ที่ได้มาต่อไป

ในทุกขั้นตอนตั้งแต่ขั้นตอนที่ 1 – 5 มีความเสี่ยงจากการลักลอบเข้าถึงข้อมูลหรือใช้งาน API โดยไม่ได้รับสิทธิ์ (Unauthorized Access) และการให้บริการหยุดชะงัก (Service Disruption) ผู้ให้บริการ API จึงให้ความสำคัญในการระมัดระวังความเสี่ยงทั้ง 2 ด้านนี้ ในทุกขั้นตอน

ภาคผนวก 2 แนวทางการประเมินความเสี่ยงตั้งต้นและการกำหนดมาตรฐานการควบคุม

หลักการพิจารณาระดับความเสี่ยงตั้งต้นเพื่อใช้เป็นแนวปฏิบัติด้านมาตรฐานการควบคุมความมั่นคงปลอดภัยสำหรับการใช้เทคโนโลยี API ในการสำหรับให้บริการทางการเงิน ประกอบด้วย 3 ส่วน ดังนี้

ส่วนที่ 1 การประเมินระดับความเสี่ยงตั้งต้น (Inherent Risk Assessment) เพื่อให้ทราบถึงลักษณะการให้บริการ API และระดับความเสี่ยงเกี่ยวกับการพัฒนา API ของตนเอง (risk profile) โดยพิจารณาจากปัจจัยความเสี่ยงพื้นฐาน 5 ด้านคือ

1. **การเปิดเผย API Specification** เป็นปัจจัยเสี่ยงที่พิจารณาถึงขอบเขตการเปิดเผย API Specification ที่ให้บริการ ประกอบด้วยขอบเขตการเปิดเผย (1) เฉพาะภายในองค์กร เช่น API Specification ที่ไม่ได้เปิดเผยให้นิติบุคคล หรือบุคคลภายนอก เป็นต้น (2) ระหว่างองค์กรที่เกี่ยวข้อง เช่น API Specification ที่มีการควบคุมการเปิดเผยต่อนิติบุคคล หรือบุคคลภายนอก เป็นต้น (3) สาธารณะ เช่น API Specification ที่ไม่มีการควบคุมการเปิดเผยต่อนิติบุคคล หรือบุคคลภายนอก เป็นต้น ซึ่งอาจก่อให้เกิดความเสี่ยงด้านไซเบอร์ในรูปแบบจากปัจจัยแวดล้อมที่แตกต่างกันตามขอบเขตการเปิดเผย เช่น การเปิดเผยค่าพารามิเตอร์ในการใช้งานและการทำงานของแอพพลิเคชันสู่สาธารณะ อาจก่อให้เกิดความเสี่ยงด้านไซเบอร์ที่สูงกว่า API Specification ที่เปิดเผยในวงจำกัด

2. **ลักษณะข้อมูลการให้บริการ** เป็นปัจจัยเสี่ยงที่พิจารณาถึงลักษณะของข้อมูลที่ให้บริการ API หมายถึงการให้สิทธิ์แก่ผู้ใช้บริการ API ในการเข้าถึงข้อมูลลักษณะต่าง ๆ ประกอบด้วย (1) ข้อมูลที่เปิดเผยต่อสาธารณะ เช่น ข้อมูลรายละเอียดของบริการ ข้อมูลช่วงเวลาที่เปิดให้บริการ ข้อมูลค่าธรรมเนียมการใช้บริการ และข้อมูลสาขาที่ให้บริการ เป็นต้น, (2) ข้อมูลที่เกี่ยวข้องกับธุรกรรมของธนาคาร เช่น ข้อมูลรายละเอียดการโอนเงิน/การชำระเงิน, ข้อมูลเกี่ยวกับข้อมูลส่วนบุคคล (PII) เป็นต้น และ (3) ข้อมูลสำคัญ คือ ข้อมูล Sensitive PII ตามที่ระบุในเอกสาร PDPA หรือข้อมูลที่มีชั้นความลับระดับสูงสุดขององค์กร เช่น ข้อมูล Biometric, PIN/Password เป็นต้น ซึ่งอาจก่อให้เกิดความเสี่ยงด้านไซเบอร์ในระดับความรุนแรงที่แตกต่างกันไปตามความสำคัญของข้อมูลแต่ละลักษณะ

3. **ความต้องการสิทธิ์สำหรับการดำเนินการ** เป็นปัจจัยเสี่ยงที่พิจารณาถึงการให้สิทธิ์การดำเนินการแก่ผู้ใช้บริการ API กล่าวถึงการกำหนดสิทธิ์ให้ผู้ใช้บริการ API (1) สามารถอ่านข้อมูลได้อย่างเดียว หรือ (2) สามารถอ่านและเขียนข้อมูลได้ และ (3) สามารถบริหารจัดการสิทธิ์ได้ เช่น การเปลี่ยนแปลงแก้ไขสิทธิ์การเข้าใช้งานอาจก่อให้เกิดความเสี่ยงด้านไซเบอร์ในรูปแบบและระดับความรุนแรงที่แตกต่างกันไปตามสิทธิ์ที่อนุญาตให้เข้าถึงการใช้บริการ API

4. **ลักษณะการเชื่อมต่อ** เป็นปัจจัยเสี่ยงที่พิจารณาถึงลักษณะการติดต่อสื่อสารหรือการเชื่อมต่อ API ผ่านเครือข่ายประเภทต่าง ๆ ทั้งภายในและภายนอกองค์กร โดยแบ่งออกเป็น 3 รูปแบบ (1) เครือข่ายภายในองค์กร, (2) เครือข่ายภายนอกองค์กรที่มีการจำกัดการเข้าถึง และ (3) เครือข่ายภายนอกองค์กรที่ไม่มีการจำกัดการเข้าถึง ซึ่งอาจก่อให้เกิดความเสี่ยงด้านไซเบอร์ในรูปแบบและระดับความรุนแรงที่แตกต่างกันไป

5. **ปริมาณข้อมูล** เป็นปัจจัยเสี่ยงที่พิจารณาถึงขนาดของข้อมูลที่เกี่ยวข้องสำหรับการใช้งาน API โดยแต่ละองค์กรสามารถพิจารณาหลักเกณฑ์การกำหนดระดับปริมาณข้อมูลตามความเหมาะสม เช่น การระบุปริมาณจำนวน Records ของ Data source หรือ ขนาดของ Data source สำหรับใช้งาน API เป็นต้น โดยปริมาณข้อมูลขนาดใหญ่อาจก่อให้เกิดความเสี่ยงด้านไซเบอร์ในระดับความรุนแรงที่สูงกว่าการให้บริการ API ที่มีปริมาณข้อมูลขนาดเล็กกว่า ซึ่งหลักเกณฑ์ดังกล่าวควรได้รับอนุมัติจากคณะกรรมการหรือผู้บริหารที่ได้รับมอบหมายการให้บริการ API

ตารางแสดงการประเมินระดับความเสี่ยงจากปัจจัย 5 ด้าน คือ การเปิดเผย API Specification, ลักษณะข้อมูลการให้บริการ, ความต้องการสิทธิ์ของการดำเนินการ, ลักษณะการเชื่อมต่อ และปริมาณข้อมูล และแบ่งผลการประเมิน เป็น 3 ระดับ คือ ต่ำ ปานกลาง และสูง มีรายละเอียดการประเมิน ดังนี้

ปัจจัย	ระดับความเสี่ยง			ผลการประเมิน
	ต่ำ	ปานกลาง	สูง	
1. การเปิดเผย API Specification	เฉพาะภายในองค์กร	ระหว่างองค์กรที่เกี่ยวข้อง	สาธารณะ	
2. ลักษณะข้อมูลการให้บริการ	ข้อมูลสาธารณะ	ข้อมูล Financial, ข้อมูลส่วนบุคคล PII	ข้อมูล Sensitive PII, ข้อมูลชั้นความลับสูงสุด	
3. ความต้องการสิทธิ์ของการดำเนินการ	อ่านอย่างเดียว (Read Only)	อ่านและเขียน (Read-Write)	สิทธิ์การบริหารจัดการ (Managing Permission)	
4. ลักษณะการเชื่อมต่อ	เครือข่ายภายในองค์กร	เครือข่ายภายนอกองค์กรที่มีการจำกัดการเข้าถึง	เครือข่ายภายนอกองค์กรที่ไม่มีการจำกัดการเข้าถึง	
5. ปริมาณข้อมูล	น้อย	ปานกลาง	มาก	

ผลการประเมินระดับความเสี่ยงตั้งต้นของแต่ละปัจจัยจะนำไปใช้ในการกำหนดแนวทางปฏิบัติตามระดับมาตรฐานความมั่นคงปลอดภัยของการพัฒนา API โดยแต่ละปัจจัยจะถูกพิจารณาด้วยหัวข้อมาตรฐานความมั่นคงปลอดภัยที่แตกต่างกันตามลักษณะความเสี่ยงของปัจจัยนั้น ๆ ที่จะอธิบายในส่วนถัดไป

ส่วนที่ 2 การกำหนดมาตรฐานด้านความมั่นคงปลอดภัยของระบบและข้อมูล

การกำหนดแนวทางปฏิบัติการควบคุมด้านความมั่นคงปลอดภัยควรอ้างอิงตามระดับความเสี่ยงตั้งต้นแต่ละปัจจัยของตนเอง เช่น ปัจจัยด้านลักษณะข้อมูลการให้บริการที่มีระดับความเสี่ยงตั้งต้นอยู่ในระดับสูงคือข้อมูลที่ให้บริการเป็นข้อมูลชั้นความลับระดับสูงสุดขององค์กร ดังนั้นแนวทางปฏิบัติตามระดับมาตรฐานความมั่นคงปลอดภัยที่เกี่ยวข้องกับปัจจัยจะพิจารณาในระดับสูงเช่นกันคือ ระดับ Advance หรือกรณีปัจจัยด้านลักษณะการเชื่อมต่อมีระดับความเสี่ยงตั้งต้นอยู่ในระดับต่ำคือ การเชื่อมเครือข่ายภายในองค์กร ดังนั้นแนวทางปฏิบัติตามระดับมาตรฐานความมั่นคงปลอดภัยที่เกี่ยวข้องกับปัจจัยจะพิจารณาในระดับต่ำเช่นกันคือระดับ Basic เป็นต้น

ระดับความเสี่ยงตั้งต้น	ระดับความปลอดภัย
ต่ำ	Basic
ปานกลาง	Intermediate
สูง	Advance

เพื่อให้ทราบถึงแนวทางปฏิบัติตามระดับมาตรฐานความมั่นคงปลอดภัย อ้างอิงตามปัจจัยและระดับความเสี่ยงตั้งต้นของการพัฒนา API ของตนเอง แต่ละปัจจัยเสี่ยงทั้ง 5 ด้านที่กล่าวมาข้างต้นมีแนวทางการควบคุมความมั่นคงปลอดภัยที่แตกต่างกันตามลักษณะของปัจจัยนั้น ๆ สำหรับแนวทางการควบคุมความมั่นคงปลอดภัย (Security Control Domains) อ้างอิงจากแนวทางที่พึงปฏิบัติทั้ง 7 ข้อของหลักการที่ 3 ดังแสดงในตารางด้านล่าง โดยพิจารณาผลการประเมินระดับความเสี่ยงของแต่ละปัจจัยเสี่ยงตั้งต้นและมาตรฐานความมั่นคงปลอดภัยเกี่ยวข้องที่ใช้พิจารณาตามแต่ละลักษณะปัจจัย เพื่อช่วยลดระดับความเสี่ยงของปัจจัยแต่ละด้านที่อาจจะเกิดขึ้น มีรายละเอียดการพิจารณามาตรฐานด้านความมั่นคงปลอดภัย ดังนี้

ปัจจัย	ผลการประเมิน	Security Control Domains																			
		Authentication			Authorization			Data Confidentiality & Integrity			Secure Communication			Secure Coding & Configuration			Audit log & Monitoring			Resource Sufficiency		
		Basic	Intermediate	Advance	Basic	Intermediate	Advance	Basic	Intermediate	Advance	Basic	Intermediate	Advance	Basic	Intermediate	Advance	Basic	Intermediate	Advance	Basic	Intermediate	Advance
1. การเปิดเผย API Specification											Secure Communication			Secure Coding & Configuration								
2. ลักษณะข้อมูลการให้บริการ		Authentication			Authorization			Data Confidentiality & Integrity						Secure Coding & Configuration			Audit log & Monitoring					
3. ความต้องการสิทธิ์ของการดำเนินการ		Authentication			Authorization												Audit log & Monitoring					
4. ลักษณะการเชื่อมต่อ											Secure Communication						Audit log & Monitoring			Resource Sufficiency		
5. ปริมาณข้อมูล														Secure Coding & Configuration			Audit log & Monitoring			Resource Sufficiency		
ค่าระดับการควบคุมความปลอดภัย *																						

*ค่าระดับการควบคุมความมั่นคงปลอดภัยแต่ละกลุ่มมาตรฐานควบคุมความมั่นคงปลอดภัย คิดจากผลของระดับสูงที่สุดที่ได้รับในแต่ละกลุ่มมาตรฐานควบคุมความมั่นคงปลอดภัยนั้น ๆ

ยกตัวอย่างเช่น ปัจจัยการเปิดเผย API Specification ต้องพิจารณามาตรฐานการควบคุมความมั่นคงปลอดภัยในหัวข้อ Secure Communication และ Secure Coding and Configuration ในขณะที่ปัจจัยลักษณะข้อมูลการให้บริการ ต้องพิจารณามาตรฐานการควบคุมความมั่นคงปลอดภัยในหัวข้อ Authentication, Authorization, Data Confidentiality and Integrity, Secure Coding and Configuration และ Audit log and Monitoring เป็นต้น ดังนั้นการกำหนดระดับแนวปฏิบัติการควบคุมความมั่นคงปลอดภัย ใช้อ้างอิงตามระดับความเสี่ยงของปัจจัยนั้น ๆ และมาตรฐานการควบคุมความมั่นคงปลอดภัยที่เกี่ยวข้องกับแต่ละปัจจัยดังตารางข้างต้น เช่น ปัจจัยลักษณะข้อมูลการให้บริการอยู่ในระดับความเสี่ยงสูง ดังนั้นควรกำหนดมาตรฐานการควบคุมความมั่นคงปลอดภัยในหัวข้อ Authentication, Authorization, Data Confidentiality and Integrity, Secure Coding and Configuration และ Audit log and Monitoring อยู่ในระดับสูงด้วย ทั้งนี้ค่าระดับการควบคุมความมั่นคงปลอดภัยแต่ละกลุ่ม คิดจากผลของระดับสูงที่สุดที่ได้รับในแต่ละกลุ่มมาตรฐานควบคุมด้านความมั่นคงปลอดภัยนั้น ๆ

ความหมายของระดับความมั่นคงปลอดภัยกับการให้บริการ API

ระดับความมั่นคงปลอดภัย	คำอธิบาย
ระดับ Basic	เป็นระดับความมั่นคงปลอดภัยขั้นต่ำ ที่มีข้อกำหนดความมั่นคงปลอดภัยในระดับพื้นฐาน เหมาะสำหรับ API ที่มีความเสี่ยงต่ำ ความมั่นคงปลอดภัยในระดับนี้ช่วยป้องกันช่องโหว่หรือภัยคุกคามระดับพื้นฐานของการพัฒนาแอพพลิเคชันที่อาจถูกค้นพบได้ง่าย หรือใช้เทคนิคที่ง่าย

ระดับ Intermediate	เป็นระดับความมั่นคงปลอดภัยมาตรฐาน ที่มีข้อกำหนดความมั่นคงปลอดภัยในระดับกลาง เหมาะสำหรับ API ที่ทำหน้าที่ให้บริการธุรกรรมทางการเงิน ธุรกรรมระหว่างธุรกิจที่มีความสำคัญ รวมถึง API ที่ทำหน้าที่ในการประมวลผลข้อมูลส่วนบุคคล หรือ API ที่เกี่ยวข้องกับกระบวนการสำคัญทางธุรกิจ ซึ่งความถูกต้องเชื่อถือได้ (Integrity) ในการทำงานเป็นปัจจัยสำคัญในการดำเนินธุรกิจ ลักษณะภัยคุกคามต่อ API ระดับนี้ ผู้โจมตีค่อนข้างมีทักษะและมีแรงจูงใจโดยมุ่งเน้นไปที่เป้าหมายเฉพาะ โดยใช้เครื่องมือและเทคนิคที่ได้รับการฝึกฝนอย่างสูงและมีประสิทธิภาพในการค้นหาและใช้ประโยชน์เพื่อหาจุดอ่อนของแอพพลิเคชัน
ระดับ Advance	เป็นระดับความมั่นคงปลอดภัยขั้นสูง ที่มีข้อกำหนดความมั่นคงปลอดภัยในระดับสูง โดยทั่วไปแล้วระดับความมั่นคงปลอดภัยขั้นสูงนี้สงวนไว้สำหรับ API ที่มีความจำเป็นต้องมีการตรวจสอบความมั่นคงปลอดภัยสูงอย่างมีนัยสำคัญ เช่น การให้บริการ API ที่เกี่ยวกับโครงสร้างพื้นฐานที่สำคัญ มีการทำธุรกรรมที่มีมูลค่าสูง ซึ่งหากเกิดข้อผิดพลาดอาจส่งผลกระทบในวงกว้างต่อภาคการธนาคาร จึงจำเป็นต้องมีระดับความเชื่อมั่นสูงสุด (High level of Trust)

ส่วนที่ 3 มาตรฐานด้านความมั่นคงปลอดภัย 3 ระดับ

Security Control Domains	Basic	Intermediate	Advance
1. Authentication มีกลไกการยืนยันตัวตนผู้ใช้งานในระดับ Application authentication เพื่อให้สามารถควบคุม และระบุตัวตนในระดับ Client หรือ User ที่เรียกใช้งานเข้ามา ป้องกันการเข้าถึงโดยไม่ได้รับอนุญาต	√	√	√
มีกลไกการยืนยันตัวตนผู้ใช้งานในระดับ Server authentication เพื่อตรวจสอบความถูกต้องของผู้ให้บริการว่าเป็นผู้ให้บริการตัวจริงก่อนเรียกใช้บริการต่าง ๆ เช่น การตรวจสอบใบรับรองอิเล็กทรอนิกส์ (Certificate)	√	√	√
มีการออกแบบการใช้งาน Token อย่างปลอดภัย (Token strength) เช่น การใช้งาน Token แบบใช้ครั้งเดียวหรือ กำหนดอายุการใช้งานของ Token แบบจำกัด (Short-lived token) รวมถึงการเลือกใช้ Strong encryption algorithm สำหรับการสร้าง Token		√	√
มีกลไกการยืนยันตัวตนโดยการใช้ Token ด้วย Digitally signed หรือจากแหล่งตรวจสอบที่เชื่อถือได้ (Authoritative source)		√	√
มีการใช้ Multi-Factor authentication เพิ่มความมั่นคงปลอดภัยของกระบวนการยืนยันตัวตน เพื่อป้องกันการเข้าถึงบัญชีผู้ใช้โดยไม่ได้รับอนุญาต			√

Security Control Domains	Basic	Intermediate	Advance
2. Authorization มีการกำหนดสิทธิ์หรือระบุขอบเขตฟังก์ชันของ API ที่อนุญาตให้ใช้งานอย่างชัดเจนตามหลักการ “Least Privilege” และ “Deny all access by default” เช่น การกำหนดสิทธิ์ให้เฉพาะอ่านข้อมูลหรือ Query อย่างเดียว	√	√	√
มีกลไกการตรวจสอบสิทธิ์การขอดำเนินการในทุกฟังก์ชันอย่างชัดเจน		√	√
การออกแบบ Token ต้องเป็นค่าที่ไม่สามารถคาดเดาได้		√	√
มีการกำหนดขอบเขตการร้องขอใช้งานในวงจำกัด เพื่อลดความเสี่ยงที่อาจเกิดขึ้นจาก Attack Surface เช่น การจำกัดเฉพาะ IP address และ Port ที่อนุญาตเชื่อมต่อ หรือการจำกัดการเชื่อมต่อด้วยอุปกรณ์ปลายทางได้รับอนุญาตเท่านั้น เป็นต้น			√
3. Data Confidentiality and Integrity มีการป้องกันการฝังข้อมูลสำคัญใน Source Code และใน Audit Log เพื่อป้องกันข้อมูลสำคัญรั่วไหล เช่น ข้อมูล Credential, ชื่อ-สกุล, เลขบัตรประชาชน เป็นต้น	√	√	√
มีการป้องกันการแสดงข้อมูลจากการประมวลผลเกินความจำเป็น (Excessive API Information) เพื่อป้องกันข้อมูลสำคัญรั่วไหล	√	√	√
มีการเข้ารหัสข้อมูลในระดับเครือข่าย (Network Layer Standard and Strong Encryption) เพื่อป้องกันข้อมูลสำคัญรั่วไหล	√	√	√
มีการเข้ารหัสข้อมูลในระดับแอปพลิเคชัน (Application Layer Standard and Strong Encryption) เพื่อป้องกันข้อมูลสำคัญรั่วไหล และการตรวจสอบความสมบูรณ์และความถูกต้องของข้อมูล (Data integrity) เพื่อป้องกันข้อมูลถูกเปลี่ยนแปลงแก้ไขจากผู้ที่ไม่ได้รับอนุญาต เช่น การทำ Hash+Salt หรือ Signing เพื่อตรวจสอบความถูกต้องของข้อมูล เป็นต้น			√
4. Secure Communication มีกลไกการเชื่อมต่อบนระบบเครือข่ายการเข้ารหัสอย่างปลอดภัยตามมาตรฐานสากล (Transport Layer Security)	√	√	√
มีการเชื่อมต่อเครือข่ายการเข้ารหัสด้วยการใช้งาน Strong cipher suite และยกเลิกการใช้งาน Weak cipher suite	√	√	√
มีกลไกการตรวจสอบใบรับรอง Certificate จากแหล่งตรวจสอบที่เชื่อถือได้ (Local Trusted CA) เช่น Certificate ที่ออกโดยผู้ให้บริการใบรับรองที่ได้รับการรับรองจากผู้ให้บริการออกใบรับรองอิเล็กทรอนิกส์แห่งชาติ (National Root Certification Authority of Thailand) หรือ Enterprise Trusted CA ที่ออกโดยหน่วยงานภายในองค์กร เป็นต้น	√	√	

Security Control Domains	Basic	Intermediate	Advance
มีกลไกการตรวจสอบใบรับรอง Certificate จากแหล่งตรวจสอบที่เป็นมาตรฐานสากล (Global Trusted CA) เช่น Certificate ที่ออกโดยหน่วยงานสากล เป็นต้น			√
มีแนวทางในการป้องกันความเสี่ยงที่เกิดจากภัยคุกคามประเภท MITM เช่น mTLS, Cert Pinning เป็นต้น			√
5. Secure Coding and Configuration มีการตรวจสอบ Input Validation ในส่วนของ Client เพื่อป้องกันการใส่ค่าไม่ตรงกับเงื่อนไข หรือ ป้องกันการโจมตีแบบ Injection	√	√	√
มีการตรวจสอบ Input Validation และ Content Type Validation ในส่วนของ Server เพื่อป้องกันการใส่ค่าไม่ตรงกับเงื่อนไข หรือ ป้องการการโจมตีแบบ Injection	√	√	√
มีการตรวจสอบการแสดงผลลัพธ์ของการประมวลผลให้อยู่ในรูปแบบที่ถูกต้อง (Output Encoding) และมีการกำหนดขอบเขตการแสดงผล เพื่อป้องการการโจมตีแบบ Injection และการเปิดเผยข้อมูลที่มากเกินไป	√	√	√
มีการจัดการการแสดงความผิดพลาดของระบบ (Error Message Handling) เพื่อไม่ให้เปิดเผยข้อมูลเชิงเทคนิคที่สำคัญของระบบ เช่น ชื่อ Field , ชื่อ Column , ชื่อฐานข้อมูล , โครงสร้างข้อมูล , โครงสร้าง Source code , ชื่อไฟล์, ชื่อฟังก์ชัน, หมายเลข IP ภายใน, ชื่อรุ่นของซอฟต์แวร์ที่ใช้ เป็นต้น	√	√	√
มีการกำหนด HTTP Request Method ที่อนุญาตให้ดำเนินการได้ และปิด HTTP Request Method อื่นที่ไม่จำเป็น	√	√	√
มีการตั้งค่า CORS (Cross-Origin Resource Sharing) อย่างปลอดภัย เพื่อป้องกันไม่ให้มีการละเมิดการเข้าถึงข้อมูลจากโดเมนอื่นที่ไม่ได้รับอนุญาต		√	√
6. Audit log and Monitoring การจัดเก็บ Log ต้องปฏิบัติตามนโยบายการเก็บรักษาข้อมูลขององค์กรและปฏิบัติตามด้วยกฎหมาย รวมถึงข้อบังคับที่เกี่ยวข้อง	√	√	√
การจัดเก็บ Log ต้องมีการควบคุมการเข้าถึง เพื่อป้องกันเปิดเผย การเปลี่ยนแปลง แก้ไข หรือทำลาย	√	√	√
การจัดเก็บ Log ต้องพิจารณาถึงความสมบูรณ์ของข้อมูลที่ถูกบันทึก เพื่อให้สามารถใช้ติดตาม ตรวจสอบร่องรอยการเข้าถึงและการใช้งานระบบหรือข้อมูลของผู้ใช้งาน รวมทั้งใช้เป็นหลักฐานตามกฎหมาย	√	√	√
มีกระบวนการหรือเครื่องมือสำหรับตรวจสอบและวิเคราะห์ข้อมูลความมั่นคงปลอดภัยของระบบเครือข่ายขององค์กร			√
7. Resource Sufficiency มีการตั้งค่า Timeout เพื่อป้องกันระบบหยุดชะงักจากทรัพยากรที่ถูกจองให้คงค้างในปริมาณมาก	√	√	√

Security Control Domains	Basic	Intermediate	Advance
มีการตั้งค่า Throttling หรือ Rate Limit เพื่อป้องกันระบบหยุดชะงักจากการถูกเรียกใช้งาน API ในปริมาณมาก		√	√
มีแนวทางในการตัดการเชื่อมต่อ (Circuit Breaker) เพื่อลดและป้องกันปัญหาจากการเกิดผลกระทบสืบเนื่อง จากปัญหาการใช้งานในปริมาณมาก เช่น การปิดบางบริการเพื่อลดปัญหาจากการเรียกใช้งานในปริมาณมาก เป็นต้น รวมทั้ง แนวทางการกลับมาเปิดให้บริการเมื่อสถานการณ์กลับมาสู่ภาวะปกติ			√
มีแนวทางในการป้องกันความเสี่ยงที่เกิดจาก API DDoS ระดับ Network layer หรือระดับ Application layer เพื่อป้องกันการหยุดชะงักของ API Server จากการถูกโจมตี			√