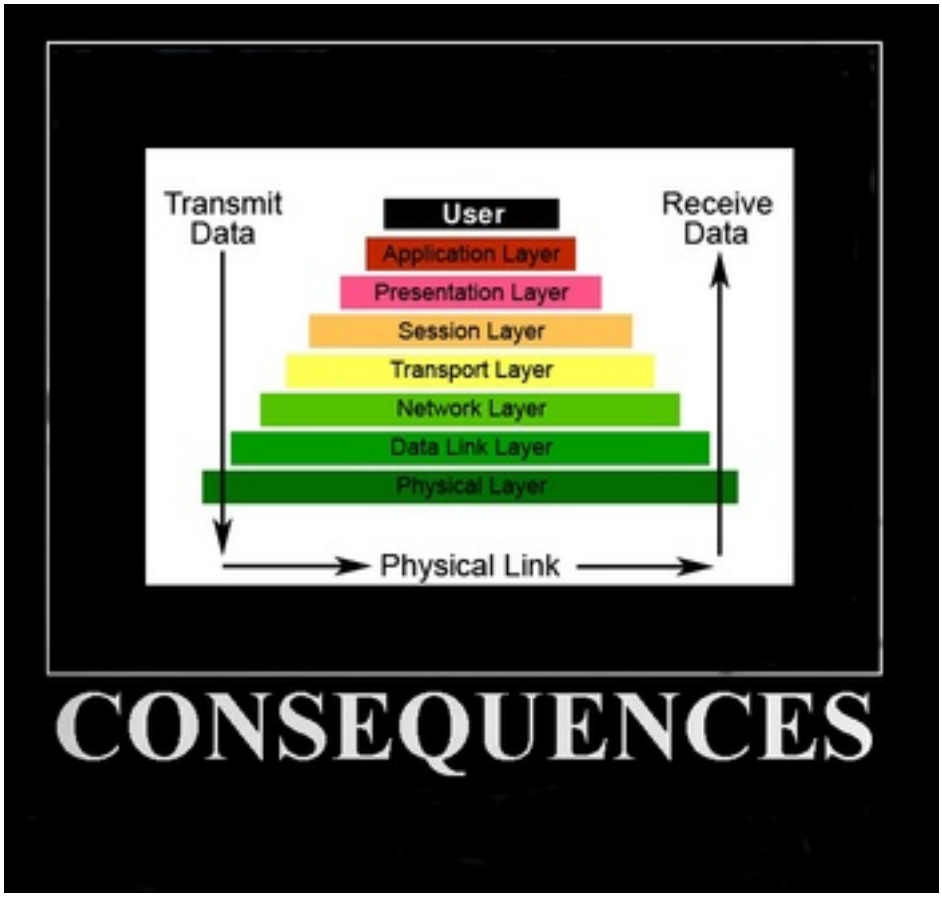
ความปลอดภัยลินุกซ์เลเยอร์ที่ 8: OPSEC สำหรับผู้ใช้ทั่วไป, นักพัฒนา และผู้ดูแลระบบ - ฉบับที่ 164 กรกฎาคม 2009

โดย Lisa Kachold **[1]**

แปลโดย Sake [2]



เมื่อผู้ใช้ของลินุกซ์แต่ละของเราอยู่ในตำแหน่งที่เฉพาะเจาะจงกับเครื่องมืออันทรงประสิทธิภาพ.
การใช้เครื่องมือใดๆโดยไม่คำนึงถึงการรักษาความปลอดภัยเป็นสิ่งที่อันตราย. ในทำนองเดียวกันที่นักพัฒนาต้องพกพารับผิดชอบอันใหญ่ยิ่งในชุมชน
เพื่อรักษาระบบที่มีให้ไปในทิศทางที่ปลอดภัย. ผู้ดูแลระบบมักได้รับตำแหน่งในบทบาทที่ไม่สบายนัก
ในการที่ต้องควบคุมป้อมปราการระหว่างความไม่มั่นคงหรือการพยามยามเข้ามาครอบครอง และช่วงเวลาในการให้บริการ.

มาลองพิจารณากันถึงหนึ่งในวิธีการการรักษาความปลอดภัยมาตรฐานเทียบกับการใช้งานลินุกซ์เป็นเครื่องมือ : OPSEC.

ปฏิบัติการรักษาความปลอดภัย (OPSEC) เป็นกระบวนการที่ระบุข้อมูลสำคัญเพื่อตรวจสอบว่าเป็นกระทำที่เป็นมิตรสามารถถูกสังเกตโดยระบบข้าศึกอัจฉริยะ,
บ่งชี้ว่าหากข้อมูลที่ได้จากข้าศึกอาจตีความเพื่อเป็นประโยชน์แก่พวกเขาแล้ว จากนั้นดำเนินการตรวจวัดที่เลือก
เพื่อจะขจัดหรือลดความเสียหายจากข้าศึกจากข้อมูลความเป็นมิตรที่สำคัญ.

"ถ้าฉันสามารถบ่งชี้แนวโน้มที่จะทำอะไรของศัตรู ขณะเดียวกับที่ฉันสามารถซ่อนตัวฉันเองแล้ว, ในตอนนั้นฉันสามารถเพ่งความสนใจและเขาจะต้องถูกแยกออกไป." - Sun Tzu

แม้ว่าเราอาจจะยังไม่ตระหนักถึงมัน, เพราะว่าเราถูกฝังรากลึกอย่างแน่นหนาอย่าง
ถึงความรู้ในเรื่อง "รากฐานความปลอดภัยลินุกซ์", ศัตรูอันเยี่ยมยอดนานาชาติ,
ระดับชาติ, และระดับท้องถิ่น ที่มีอยู่ ผู้ที่กำลังหาประโยชน์มีความสุขกับ TCP/IP Stack
ขณะเดียวกับที่หัวเราะอย่างคลั่งไคล้. ถ้าคุณไม่เชื่อฉัน, คุณใช้บรรทัดฐานอะไรในการโต้แย้งกรณีของคุณ?
คุณได้เคยลองทดสอบ หรือนำกระบวนวิธีการประเมิน OPSEC ไปใช้กับ (เลือกอย่างหนึ่ง):

```
a) ???????? SSH
b) ???????????????????? or db2/mysql/JDBC
c) ??????????? (SMTP, DNS, ????, LDAP, SSH, VPN)
```

OPSEC เป็นวิธีการได้รับการพัฒนาระหว่างสงครามเวียดนามเมื่อเรือโท Ulysses Sharp, ผู้บัญชาการในหัวหน้าแปซิฟิกจัดตั้งทีม "มังกรม่วง"
หลังจากที่ตระหนักว่า การป้องกันอัฉริยะ และการวัดความปลอดภัยแต่เพียงอย่างเดียว ไม่เพียงพอ.
พวกเขาเข้าใจและใช้กระบวนวิธีของ "คิดอย่างหมาป่า" ให้เป็นประโยชน์.
หรือมองค์กรของคุณในมุมมองที่ผู้บุกรุกดู. เมื่อการพัฒนาและการแนะนำการกระทำที่ถูกต้องไปยังการสั่งการ, พวกเขาได้สร้างสิ่งที่เรียกว่า
"ความปลอดภัยในการดำเนินงาน"(Operations Security).

OPSEC เป็นความเชี่ยวชาญในการประเมินที่ดีที่สำคัญอย่างมากอย่างหนึ่ง ที่จะสอนให้กับผู้ที่กำลังเรียนรู้
ที่จะไว้วางใจอย่างเหมาะสม และอยู่ในโลกของ "สุนัขกินสุนัข" อันใหญ่ยิ่งนี้.
นักจิตวิทยาเคนแนะนำกับฉันในครั้งหนึ่งว่า "การคิดในทุก ๆ สิ่งที่คุณควรทำ(แต่ไม่ได้ทำ) " เป็นเทคนิคที่ประเมินค่าไม่ได้
ในการทำความเข้าใจมนุษยชาติ, ธรรมชาติ,แรงบันดาลในส่วนตัว และความยุงเหยิง/เป็นระเบียบในระบบธรรมชาติ.
เมื่อชาวลินุกซ์มีแนวโน้มที่จะสนใจในการคำนวณ, เป็นผู้นำเทคโนโลยีที่มีดวงตาแวววาว, และ
และคำตอบนั้นจะเป็นผล - พวกเขาต่างกส่วนมากก็มีจรรยาบรรณอันสูงส่ง และมีความรับผิดชอบอยางเป็นเหตุเป็นผล
ต่อความปลอดภัยของระบบคอมพิวเตอร์, เมื่อพวกเขารู้ว่าจะเริ่มจากที่ใด.

บัดนี้, เราต่างก็รู้เราว่าความปลอดภัยคอมพิวเตอร์เป็นกระบวนการแบบชั้น, อย่างไรก็ตาม เรา, ในฐานะผู้ใช้,
ผู้พัฒนา, และผู้ดูแลระบบ ก็จะสร้างชั้นเพิ่มมาอีกหนึ่งชั้น.

โมเดล OSI เป็น โมเดลชั้นนามธรรม 7 ชั้นที่อธิบายสถาปัตยกรรมของการสื่อสารข้อมูลสำหรับคอมพิวเตอรที่ถูกต่อกันเป็นเครือข่าย.
ชั้นที่ถูกสร้างอยู่เหนือชั้นอื่น ๆ จะอนุญาตสำหรับให้เฉพาะบางฟังก์ชันในเชิงนามธรรมในแต่ละชั้น. ชั้นบนสุด(ชั้นที่7)
เป็นชั้นแอพพลิเคชันที่อธิบายกระบวนวิธี และโปรโตคอลของแอพพลิเคชันซอฟต์แวร์.

ชั้นที่ 8 เป็นศัพท์เฉพาะอินเตอร์เน็ตที่ถูกใช้เพื่ออ้างถึง "ผู้ใช้" หรือ "ในทางการเมือง".
ชั้นที่ถูกเพิ่มเติมมาจากโมเดล OSI ของเครือข่ายคอมพิวเตอร์.

เมื่อตัวเลขของชั้น OSI เป็นสิ่งที่ถูกพูดถึงโดยทั่วไปเกี่ยวกับประเด็นด้านเครือข่าย.
ผู้เชี่ยวชาญในการแก้ปัญหาหนึ่งอาจจะอธิบายประเด็นหนึ่ง ที่เกิดขึ้นจากผู้ใช้ไเป็นประเด็นของชั้นที่ 8.
เช่นเดียวกับตัวย่อ PEBKAC (Problem Exists Between Keyboard And Chair) และ ID-Ten-T Error (คำที่นักแก้ปัญหาใช้แทนปัญหาที่เกิดจากผู้ใช้).

เราสามารถเห็นได้ว่ากุญแจ SSH (หรือการไม่มีสิ่งเหล่านี้) แต่เพียงอย่างเดียวไม่เป็นประเด็นด้านความปลอดภัยที่ใหญ่อะไร.
อย่างไรก็ตาม เพิ่มผู้ใช้ root, ที่อยู่อินเตอร์เน็ตที่สามารถหาเส้นทางได้อย่างเต็มที่ (นอกจาก NAT),
ไม่มี iptables หรือไฟร์วอลล์อื่น ๆ, ไม่มีการจัดการรหัสผ่าน หรือนโยบายทางด้านความปลอกภัย และ
การดักจับอีเธอร์เน็ตแปลกปลอมจากผู้ใช้ติดอาวุธที่ดกรธแค้น ที่มีบุคลิกลักษณะที่ผิดปกติ ไม่เข้าพวก. เอาล่ะ,
สิ่งเหล่านี้อาจจะเป็นปัญหาหรือไม่?

ขั้นตอนการประเมิน OPSEC ได้แก่:

1. ระบุข้อมูลสำคัญ

ระบุข้อมูลที่สำคัญอย่างอย่างยิ่งกัยองค์กร, ภารกิจ, โครงการ หรือบ้าน [กรรมสิทธิ์ที่สำคัญ, รายละเอียดภารกิจ, ขีดความสามารถในการวิจัยและพัฒนา, การเสื่อมเสียชื่อเสียง, ข้อมูลการใช้งานของเจ้าหน้าที่ที่สำคัญ, บันทึกทางการแพทย์, นิติกรรมสัญญา, แผนผังโครงสร้างเครือข่าย, อื่น ๆ.]

แต่ "เดี๋ยวก่อน", คุณบอก, "ฉันเป็นแค่เด็กเล็ก ๆ กับคอมพิวเตอร์แลปท็อป!" คุณไปร้านขายกาแฟเป็นประจำเลยหรือเปล่า?
คุณใช้เครือข่ายร่วมกับผู้อื่นหรือเปล่า? คุณอนุญาตให้คนอื่น ๆ ดูคุณเข้าใช้และไปดูข้อมูลธนาคารของคุณจากด้านหลังของคุณหรือเปล่า?
ถ้าเช่นนั้น, OPSEC มีเพื่อคุณอย่างไม่ต้องสงสัย.

ขณะที่ขั้นตอนนี้ เป็นมีความเยี่ยมยอดมากเท่ากับความเลวร้ายสำหรับที่จะเป็นผู้ดูแลระบบ, ผู้ที่รู้อย่างชัดแจ้งว่า ง่ายแค่ไหน ที่จะระเบิดระบบหนึ่ง และทิ้งงานทั้งหมดจากมืออาชีพ 30 คนหรือมากกว่าด้วยคำสั่งเพียงคำสั่งเดียว, หรือเป็นอีกผู้หนึ่ง ซึ่งตระหนักว่าความเป็นเจ้าของระบบ ไม่ได้หมายถึงสิ่งที่จะบอกได้ถึงอัพไทม์อันมั่นคง, แต่ละ และทุก ๆ ผู้ใช้คอมพิวเตอร์รู้ว่า อะไรที่มันเป็นไม่ได้ขึ้นอยู่กับความมั่นคงของข้อมูลใด ๆ. ความปลอดภัย = ความมั่นคงในทุก ๆ สถานะการณ์.

2. ระบุศักยภาพศัตรู

ระบุศัตรูที่เกี่ยวข้อง, คู่แข่งหรืออาชญากรรวมทั้งความมุ่งหมาย และความสามารถในการได้มา ถึงข้อมูลสำคัญของคุณ.

หากคุณยังไม่เพิ่งดำเนินการเพื่อดูล็อกไฟล์ของคุณสักครู่. เพื่อดูทั้งหมด พยายามเข้าใช้งานผ่าน SSH หรือ FTP, หรือนั่งอย่างจริงจัง
ในร้านกาแฟประเมินอัตราการเข้าชมเครือข่ายไร้สาย หรือจับตาดู tcpdump เพื่อดู สิ่งที่อยู่ที่เกิดขึ้นในเครือข่ายมหาวิทยาลัย นี่อาจถึงเวลาที่จะ เริ่มต้น.
มันไม่ใช่แค่คนที่มาจากจีนและรัสเซีย (รวมถึงสคริปต์ netcat, nmap และ MetaSploit ที่สามารถถูกกำหนดค่านิด ๆ หน่อย ให้ตบตา ที่อยู่เหล่านี้) ตื่น
และกวาดตาไปยังที่ประชุม และถาม
เองอย่างจริงจัง, ใครที่เป็นคู่แข่ง, ใครที่เป็นอาชญากร. นี่คือขั้นตอนที่จำเป็นใน OPSEC. ในปี 1990 ในแปซิฟิกตะวันตกเฉียงเหนือ, ผู้ดูแลระบบลินุกซ์ฝ่ายตรงข้าม
กำลังโจมตีเว็บเซิร์ฟเวอร์ของคนอื่น ๆ.

แต่, คุณพูดว่า, "ทำไมพวกเขาจะต้องการเข้ามายังแล็ปท็อปน้อยของฉันด้วย?". คุณมีอัพไทม์ 24x7 ถูกต้องมั้ย? ระบบของคุณ สามารถถูกตั้งค่าได้ด้วย Anacron
เป็นสิ่งที่ไม่แน่ชัด. BOT net ที่ตรวจสอบไม่ได้และคุณไม่เคยรู้เกี่ยวกับมันมาก่อน? BOT net นั้น สามารถเขมือบแบนด์วิทของคุณและขโมยพลังการประมวลผล
และในที่สุดจะถูกใช้เพื่อเอาเซิร์ฟเวอร์คุณลงมา.

3. ระบุช่องโหว่ที่อาจเกิดขึ้นได้

ในมุมมองของศัตรู, สู่แข่ง, หรือโจร, ระบุช่องโหว่ที่อาจเกิดขึ้นได้ และความหมายในการเข้าถึงไปยังผลของขั้นตอนที่ 1. ทำการซักถามตัวแทนตัวอย่างของแต่ละบุคคล.

ถ้าคุณยังไม่ได้ค้นกูเกิ้ล เพื่อที่จะแน่ใจว่ารุ่นของ Firefox ที่คุนกำลังให้อยู่ปลอดภัยจากการบุกรุกหาประโยชน์แล้วละก็.
คุณก็ยังไม่ทำขั้นตอนนี้ให้สมบูรณ์. ถ้าคุณยังไม่รู้ถึงความอ่อนแอ ณ ปัจจุบันของรุ่นของ OpenSSH และ Apache หรือ Java
หรือ
If you have not googled to ensure that the version of Firefox you are running
is secure from known exploits, you have not completed this step. If you don't
know the current vulnerabilites of the version of OpenSSH and Apache or Java
หรือการถูกแก้ไขของรหัสต้นฉบับของไบนารีที่ถูกติดตั้งหับ Ubuntu หรือ Fedora ของคุณ,
คุณยังไม่มีพื้นฐานการใช้เทคโนโลยีลินุกซอย่างชาญฉลาด.

แม้ว่าฉันจะไม่แนะนำให้ทุก ๆ คนเข้าร่วมกับ Defcon, การอ่านเว็บไซต์ของพวกเขาอาจจะ
เพียงพอที่จะเน้นย้ำคุณสนใจถึงความสำคัญของ OPSEC. แต่จะดีกว่า, ถ้าคุณเขียนแผ่น BackTrack4 LiveCD ของคุณเอง
และเรียกใช้เครื่องมือเพื่อป้องกันของระบบของคุณเอง.

มีสองวิธีพื้นฐานที่จะเข้าไปใช้ Linux โดยการใช้ระดับชั้นโมเดล OSI :
"บนลงล่าง" หรือ "ล่างขึ้นบน".

จาก: โมเดล OSI [3]

ชั้น :

1. ชั้นรูปธรรม (Physical Layer)
2. ชั้นสื่อสารข้อมูล (Data Link Layer)
    - สถาปัตยกรรมโปรโตคอล WAN
    - สถาปัตยกรรม IEEE 802 LAN
    - สถาปัตยกรรม IEEE 802.11
3. ชั้นเครือข่าย (Network Layer)
4. ชั้นขนส่ง (Transport Layer)
5. ชั้นวาระ (Session Layer)
6. ชั้นนำเสนอ (Presentation Layer)
7. ชั้นโปรแกรมประยุกต์ (Application Layer)
8. ชั้นมนุษย์ (Human Layer)

ประเด็นส่วนใหญ่อยู่ที่ชั้นที่ 8!

4. ประเมินความเสี่ยง

ประเมินความเสี่ยงของแต่ละความไม่มั่นคง โดยผลกระทบของสิ่งนั้น ๆ ต่อการบรรลุภารกิจ / ประสิทธิภาพเมื่อต้องการ

หลังจากทดสอบ SSH ของคุณจากเครือข่ายถายนอก หรือสแกนกลุ่มของโปรแกรมประยุก J2EE ของคุณ
จากรหัส IBM WatchFire AppScan, หรือ
After testing your SSH via an outside network or scanning your J2EE
การกวน Apache 1.33/LDAP จากการใช้ร่วมกัน (ไม่มี VLAN เครือข่ายสาธารณะ) หรือการเข้าถึง/การเจาะ
รหัส WEP ของคุณเองในห้านาที, คุณจะเห็นอย่างชัดเจนว่า คุณไม่จะเป็นที่จะต้องมีอะไรเลย,
สามารถพิสูจน์ถึงความไม่มีเสถียรภาพ, และทันทีที่ระบบของคุณถูกละเมิดจะถูกปิดลง.

ตัวอย่างเช่น, จุด ที่ผู้ใช้ /ผู้ดูแลเซิร์ฟเวอร์ iPhone/Blackberry ตระหนักคือ โทรศัพท์ของเขา
บางทีจะทำอะไรมากกว่าที่เขาคาดคิดได้ และสงสัยเมื่อไม่ได้ตรวจสอบ , ไม่ได้สแกน นโยบายการแนบเข้ามาของไฟล์ pdf ตั้งค่าในชั้นที่ 8/9,
แต่โดยปราศจาก OPSEC, จุดนี้มักเกิดขึ้นซ้ำไปตลอด. โทรศัพท์มีศกยภาพที่จะควบรวมเข้ากับทุกอย่าง, มีที่อยู่ IP และมักจะถูกเพิกเฉย!

อีกครั้ง, ตัวอย่างเช่น, ถอดเครื่องพิมพ์ออก, เป็นสิ่งที่โง่เขลา, เนื่องจากเครื่องพิมพ์ HP และโปรโตคอล IPP
ไม่มีความสำคัญที่จะไปทำการละเมิด หรือปล่อยเชื่อกับอะไรซักอย่างจากสิ่งที่ปลอดภัยที่แนบมา.

ขั้นตอนนี้ ครอบคลุมรวมถึงทุก ๆ เทคโนโลยีที่เกี่ยวข้องกัน ที่ระบบติดต่อประสานงานด้วย.

5. กำหนดมาตรการการต่อต้าน

สร้าง / ระบุการมาตรการที่จะต่อต้านที่จะระบุความอ่อนแอ. ให้ระดับความสำคัญและออกมาตรการที่เกี่ยวข้องกับการป้องกัน.

ตอนนี้ ก่อนที่คุณจะตกใจสุดขีดและกรีดร้อง, เริ่มกระบวนการนี้ สาเหตุจากความเสียหายจาก "ข้อจำกัด"
จากอิสระของการประมวลผล, จงมั่นใจว่า นั้นคือ "คำตอบ".

6. ประเมินผลประสิทธิผล

ประเมินและวัดประสิทธิผล,และปรับปรุงตามนั้น.

คำตอบเหล่านี้ อาจจะง่าย ๆ พอกับการห่อหุ้มสำหรับ SSH, การปรับปรุงไปใช้ OpenVPN
(ซึ่งเป็นเรื่องธรรมดา ๆ ในการทำให้เกิดขึ้น). การใช้ noscript สำหรับบราวเซอร์,
หรือการไม่ใช้เว็บบราวเซอร์จาก root ในเครื่องที่ทำงานจริง. สิ่งเหล่านี้สามารถรวมใน
เครื่องเซิร์ฟเวอร์ที่พื้นฐานบน IPTABLE ที่ติดตั้งครั้งแรก, หรือรวมอยู่ในสวิตซ์โปรแกรมประยุกต์ชั้นที่ 7
(ซึ่งสามารถจะถูกกว่าในร้านการพัฒนา "house of card" แทนที่จะเป็นการตรวจสอบโค้ดทั้งหมด
(สำหรับที่เข้ากันได้กับ PCI)) หรือ เขียนใหม่หมด. สำหรับกรณีการก่อกวน Tomcat. การปรับเปลี่ยน
สถาปัตยกรรมด้วยการกั้นด้วย IDS หรือใช้ mod_security หรือ mod_proxy สามารถจะประหยัดเวลาได้เป็นเดือน ๆ สำหรับการ DoS

ถ้าคุณจะต้องเข้าร่วมกับเครือข่ายทางสังคม หรือท่องไปยังเว็บไซต์ที่ต้องระวังการยอมรับ javascript,
การเข้าจากระบบกึ่งไม่ปลอดภัยอาจจะเป็นมาตรการที่ดี.

ลองคิดนอกกรอบดูว่า คำตอบแบบระดับชั้นเป็นสิ่งสำคัญ
และอย่างน้อย, ออกทันทีทันใด จากการกระทำที่บ่งชี้ว่าเป็นอันตรายที่ระดับชั้นที่ 8
ปิด SSH ที่ร้านขายกาแฟ. ในความเป็นจริง, ปิดบลูทูธด้วนเป็นการดี เพราะการเชื่อมต่ออาจจะยัง
คงมีอยู่จากการที่คุณสร้างขึ้นหรือเปล่า?

การจัดการที่ดีของข้อมูลไม่ครอบคลุมยังการสืบสวนนี้,
ดังนั้นเอกสารที่ถูกจัดการการเข้าถึงจะทำให้คุณสามารถจัดเรียงมันได้.
คุณจะพบว่ามันอาจจะง่ายกว่าที่จะแทนที่หรืออัพเกรด, แทนที่จะพยายามป้องกันมัน
มันไม่เป็นความจริงเท่าไหร่ ที่จะคาดหวังว่าจะต้องอัพเกรดอย่างน้อยทุก ๆ สี่ปี,
ลองพิจารณาดูจะเห็นว่า คุณทำการใช้งานตัวปรับปรุงมาตรฐานซึ่งผ่านมากว่า
10 ปีแล้วของประวัติศาสตร์ลินุกซ์.

ถ้า (เมื่อ) คุณบ่งชี้ถึงอันตรายของนโยบาย, เอกสาร และการเพิ่มขึ้นของชั้นที่ 9 (การบริหารจัดการ),
ดงนั้น หน้าที่ของคุณในฐานะผู้ใช้หรือผู้ดูแลระบบเทคโนโลยีที่มีจรรยาบรรณ ได้ถูกกส่งต่อไปหรือขึ้นไป.
การขาดความกระตือรือร้นหรือทัศนคติอย่า "การรายงานด้านความปลอดภัยเป็นสิ่งที่เค้าไม่ทำกันหรอก" หรือ
"ทุกคนเค้าไม่สนใจไอ้ยาว ๆ นี่หรอก, ขืนฉันบอกไปบางทีเค้าอาจจะมองว่าฉันไร้สาระ" เป็นสิ่งที่ทำลายทุกอย่าง.
ใช้การอ้างอิงถ้าจำเป็น; ความปลอดภัยเป็นหน้าที่ความรับผิดชอบของทุกคน.

อย่างหนึ่งที่ของนโยบายที่อันตรายระดับชั้นที่ 8/9 อย่างหนึ่งคือการแบ่งระดับ.
ตัวอย่างหนึ่งที่สำคัญของการแบ่งระดับนี้ เช่น :

- Ubuntu ปลอดภัยมาตั้งแต่ในกล่องแล้ว
- Linux ปลอดภัยกว่า Windows, ดังนั้นฉันไม่เห็นต้องกังวลอะไร
- เจ้าหน้าที่ความปลอดภัยเรามี ไม่ใช่หน้าที่ของฉันนิ
- แผนกไฟร์วอลล์ของเราดูทุก ๆ ประเด็นของความปลอดภัยอยู่แล้ว, ไม่ใช่หน้าที่ของ webmaster

เมื่อสุขภาพทีดีของความรับผิดชอบต่อสภาพแวดล้อมด้านความปลอดภัยในองค์กร ย้ายไปใน
สุขภาพทีดีของความรับผิดชอบต่อสภาพแวดล้อมด้านความปลอดภัยในซอฟต์แวร์เปิดรหัส เป็นกระบวนการที่ถูกจัดขึ้น,
และความปลอดภัยทุก ๆ ที่จะไม่สามารถเพิกเฉยได้.

ความอ่อนแอเป็นศูนย์เป็นที่เป็นไปไม่ได้เลยในความเป็นจริง.
ถ้าคุณทำงานในร้านที่การทำงานเหมือนว่าเป็นระบบความปลอดภัยสมบูรณ์แต่ไม่มี OPSEC
หรือติดต่อตัวคุณเองเหมือนกับว่ำไม่มีความเสี่ยงด้านความปลอดภัย.
เป็นสิ่งที่ต้องปักธงแดงเอาไว้เลยว่า.
การเฝ้าระวัง 100% เป็นสิ่งที่พยายามจะปฏิบัติให้ได้จริง.

ตามกฎทั่วไป, การระวังสิ่งที่จะปกป้องมีความเสี่ยงมากกว่าในการป้องกัน
ข้อมูลที่อ่อนไหวต่อการเปลี่ยนแปลง ตรงกันข้ามกับการไม่ระวังค่านั้น.

รับข้อมูลการคุกคามจากผู้เชี่ยวชาญ, อย่าพยายามจะวิเคราะห์ทุกอย่างด้วยตัวคุณเอง.

เนื่อาจจะเป็นการค้นรายการข้อมูลแบบง่าย ๆ จาก CERT ที่เกี่ยวกับเทคโนโลยีที่คุณใช้,
อาจเป็น Cisco IOS สำหรับ Pix ของคุณ หรือแค่ OpenWRT.

และมากกเกินกว่าที่คุณจะต้องการจะเชื่อ, โดยมากแล้ว
ยังคงมี 8% ของยังสามารถการบุกรุกผู้เชียวชาญความปลอดภัยคอมพิวเตอร์ได้,
หลังจากคุณปรับลดความเสี่ยงที่เกิดขึ้นทั้งหมดแล้ว, ก็ยังคงเข้าสู้ระบบคุณได้.
คุณสมบัตินี้เป็นจริงสำหรับการวิเคราะห์ของคุณ.

สนใจว่าอะไรที่จะปกป้องได้และอะไรที่เพิ่งถูกเปิดเผย.

ตัวอย่างเช่น, คุณอาจตระหนักว่า หลังจากที่เล่นกับ BackTrack3 SMB4K
และเวลานี้ SAMBA ของคุณได้อนุญาตการแบ่งปันไฟล์ไร้สายไปยัง Windows เพื่อนบ้าน
และข้อมูลส่วนตวของคุณ รวมถึงรูปส่วนตัว มีอยู่กับทุกคน. เพียงแค่ปิดมันซะ.
การหวาดระแวงในความปลอดภัย เป็นตัวเลือกหนึ่ง แต่ไม่แนะนำเป็นอย่างยิ่ง.

สังเกต, ค้นหา และ เสนอมาตรการการต่อต้าน
เป็นรูปแบบในแผนการปฏิบัติกับเหตุการณ์สำคัญ
ที่จะทำให้ความอ่อนแอลดน้อยลง
ในสภาวะการทำงานจริง, แผนนี้ควรที่จะถูกส่งต่อในการสร้างการตัดสินใจโดยย่อให้สมบูรณ์,
กับทีมผู้ใช้ที่คิดต่อกัน และคนอื่น ๆ เป็นเสาหลักในอัพไทม์.

ผสมผสาน OPSEC เข้ากับการวางแผนและกระบวนการตัดสินใจตั้งแต่การเริ่มต้น.
การรอคอยจนกระทั่งนาทีสุดท้ายก่อนที่จะพูดถึงผลิตภัณฑ์ไปยังตลาด (หรือจากตลาด)
เพื่อที่จะควบคุมการประเมินค่าอาจจะช้าเกินไปและสิ้นเปลือง.

"อะไรคือแผนก QA?" คุณอาจจะถาม. "เราคือแผนก QA และเราไม่มีเวลาที่จะสแกน?"
มันง่ายที่จะใช้ Wikto/Nikto หรือใช้การสแกนเพื่อต่อต้านโปรแกรมของคุณ.

หาที่ดี ๆ อื่น ๆ ให้กับ php/Mysql CMS หรือ ที่ว่างแบ่งอื่น ๆ? ใช้ SVN?
คุณอาจเพิ่งจะเปิดท่อที่เข้ารหัสที่ดีอันนึงโดยตรงไปยังระบบของคุณ.
ถ้าคุณไม่ทดสอบมัน, คุณจะไม่รู้! คุณเคยทดสอบรุ่นของ SugarCRM ก่อนทำการ build หรือไม่?
คุณมองไปยังพฤติกรรมการบุกรุกที่รู้จักของเครื่องมือแบบเปิดรหัสก่อนรับมมาหรือไม่?

ประเมินค่าปกติเพื่อแน่ใจการป้องกันที่ดีของคุณ.

เหมือนการกู้ภัยจากภัยพิบัติ, การประเมินค่าระบบ OPSEC ได้สร้างมาจากแบบนั้น.
ข้อมูลที่เปิดเผยจะถูกเก็บไว้เป็นแหล่งที่มา; การประเมินผลแยยมีกำหนดการเปนประจำอย่างต่อเนื่อง
เป็นเหตุการณ์กลุ่มที่ทำงานพร้อมกัน.

ความปลอดภัยของระบบไม่ใช้ความลับ; OPSEC เตือนเราว่า ระบบทั้งหมดเพียงแต่ป่วย เช่นเดียวกับความลับของพวกเขา.

Talkback:
พูดคุยเรื่องบทความนี้กับ The Answer Gang [4]

Lisa Kachold เป็นผู้ดูแลระบบ/ความปลอดภัยบนลินุกซ์, ผู้ดูแลเว็บ,
inactive CCNA, และผู้เขียนโปรแกรม และกว่า 20 ปีกับประสบการณ์ในการใช้งานจริงกับลินุกซ์.
Lisa ผ่านการเป็นครูจาก FreeGeek.org, นักนำเสนอที่
DesertCodeCamp, ผู้ใช้ Wikipedia and สมาชิก LinuxChix.
เธอจัดการและประชาสัมพันธ์การศึกษาความปลอดภัยบนลินุกซ์ ไปยัง Phoenix Linux Users
Group HackFEST Series labs, ใช้สองเสาร์ของทุก ๆ เดือนที่ The
Foundation for Blind Children in Phoenix, Arizona. Obnosis.com, a play
on a words coined by LRHubbard, ลงทะเบียนใน in the 1990's, เป็น "word
hack" จาก the Church of Scientology, หลังจาก 6 ปีของผู้ดูแลข่าว UseNet.
ความภูมิใจที่สุดของคือคือการที่ได้นั้งกับ Linux Torvald's
ระหว่างการสัมภาษณ์ที่ OSDL.org ใน Oregon ในปี 2002.

ตีพิมพ์ในเล่มที่ 164 ของ of Linux Gazette, กรกฎาคม 2009

-
- อ่าน 4122 ครั้ง

Source URL (modified on 2009-08-15 10:37): https://sake.in.th/node/81

Links

[1] http://linuxgazette.net/authors/kachold.html
[2] http://sake.in.th
[3] http://en.wikipedia.org/wiki/OSI_model
[4] mailto:tag@lists.linuxgazette.net?subject=Talkback:164/kachold.html
[5] http://linuxgazette.net/copying.html