



ຄະນະວິທະຍາສາດທຳມະຊາດ  
ພາກວິຊາ ວິທະຍາສາດຄອມພິວເຕີ

# ຄວາມປອດໄພເວບໄຊ້ (Web Security)

ສອນໂດຍ: ອຈ ເພັດ ສອນວິໄລ

ມືຖື: 020 58390300

ອີເມວ: [p.sonevilay@nuol.edu.la](mailto:p.sonevilay@nuol.edu.la)



# ບົດທີ 1

## ຄວາມຮູ້ເບື້ອງຕົ້ນກ່ຽວກັບຄວາມປອດໄພ (Web Security Fundamentals)

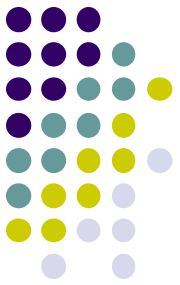




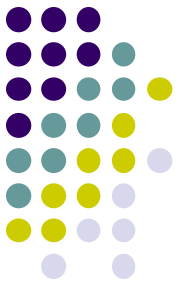
# ເນື້ອໃນໂດຍລວມ

- ສະເໜີເບື້ອງຕົ້ນ
- ນິຍາມຂອງຄວາມປອດໄພ
- ອົງປະກອບພື້ນຖານຂອງຄວາມປອດໄພ
- ໄພຄຸກຄາມ
- ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ
- ສະຫຼຸບ

# ສະເໜີເບື້ອງຕົ້ນ



- ຄວາມປອດໄພຂອງເວບໄຊ້, ມີຫຼາຍແນວຄວາມຄິດ ເຊັ່ນວ່າ:
  - ການບຸກໂຈມຕີເວບໄຊ້(defacing Web sites).
  - ລັກລະຫັດບັດ credit(card numbers)
  - ຄຸກຄາມເວບໄຊ້ດ້ວຍການສົ່ງປິດບັນດາບໍລິການໃດໜຶ່ງບໍ່ໃຫ້ ເຮັດວຽກໄດ້(bombarding Web sites).
  - ໄວຣັດ(viruses), Trojan horses ແລະ Worms
- ເວບໄຊ້ໃນປັດຈຸບັນພົບບັນຫາດັ່ງກ່າວຄຸກຄາມ.
- Firewall ບໍ່ສາມາດປ້ອງກັນໄດ້ໝົດ.



# ນິຍາມຂອງຄວາມປອດໄພ

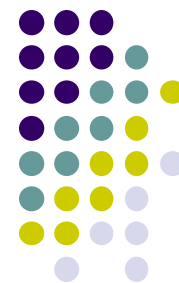
- ຄວາມປອດໄພ ເປັນພື້ນຖານການປ້ອງກັນຂໍ້ມູນຊັບສິນ, ຊຶ່ງຊັບສິນດັ່ງກ່າວລວມທັງສິ່ງທີ່ຈັບຕ້ອງໄດ້ ເຊັ່ນວ່າ: ເວບເພຈ (Web page), ຖານຂໍ້ມູນເກັບກຳຂໍ້ມູນລູກຄ້າ ຫຼື ຂໍ້ມູນຂອງອົງກອນຕ່າງໆ.
- ຄວາມປອດໄພຄືກັບເສັ້ນທາງທີ່ບໍ່ມີຈຸດສິ້ນສຸດ, ຊຶ່ງຜູ້ຮັກສາຄວາມປອດໄພຕ້ອງໄດ້ວິເຄາະໂຄງສ້າງລະບົບ ແລະ ໂປຣແກຣມທີ່ນຳໃຊ້ຢູ່ ເພື່ອໃຫ້ຮູ້ໄດ້ເຖິງຄວາມສ່ຽງ, ຈຳແນກໄດ້ຄວາມສ່ຽງແຕ່ລະໆດ້ບ ແລະ ວາງແຜນບໍລິຫານຈັດການກັບຄວາມສ່ຽງດັ່ງກ່າວໃຫ້ສາມາດຄວບຄຸມໄດ້.



# ອົງປະກອບພື້ນຖານຂອງຄວາມປອດໄພ

- ຄວາມປອດໄພຂອງຂໍ້ມູນປະກອບມີດັ່ງນີ້:
  - ການຢືນຢັນຜູ້ໃຊ້ (**Authentication**)
  - ການອະນຸຍາດ(**Authorization**)
  - ການກວດສອບ (**Auditing**)
  - ຄວາມລັບຂອງຂໍ້ມູນ(**Confidentiality**)
  - ຄວາມຖືກຕ້ອງຂໍ້ມູນ(**Integrity** )
  - ຄວາມພ້ອມໃຫ້ບໍລິການຂໍ້ມູນ(**Availability** )

# ອົງປະກອບພື້ນຖານຂອງຄວາມປອດໄພ



- ການຢືນຢັນຜູ້ໃຊ້ (**Authentication**)
  - ການຢືນຢັນຜູ້ໃຊ້ເປັນການຖາມຄໍາຖາມຜູ້ໃຊ້ລະບົບເພື່ອຢືນຢັນວ່າຜູ້ໃຊ້ດັ່ງກ່າວມີຕົວຕົນຢູ່ໃນລະບົບບໍ່ ເຊັ່ນວ່າ: ຖາມຄໍາຖາມຜູ້ໃຊ້ວ່າ: “ເຈົ້າແມ່ນໃຜ(who are you)?” ຂະບວນການນີ້ຈະເປັນການຈຳແນກໃຫ້ຮູ້ໄດ້ລະຫວ່າງຜູ້ໃຊ້ໂປຣແກຣມ(Clients) ແລະ ການໃຫ້ບໍລິການ, ຊຶ່ງຜູ້ໃຊ້ໂປຣແກຣມ(Clients) ເຫຼົ່ານີ້ ອາດຈະເປັນຜູ້ທົ່ວໄປ(End user), ບໍລິການອື່ນໆ, ຂະບວນການ ຫຼື ຄອມພິວເຕີ ກໍເປັນໄປໄດ້.
  - ສໍານວນ “ການຢືນຢັນຜູ້ໃຊ້” ຢູ່ໃນຄວາມໝາຍຄວາມປອດໄພແລ້ວເອີ້ນວ່າ: ຂໍ້ກຳນົດ (Principals)

# ອົງປະກອບພື້ນຖານຂອງຄວາມປອດໄພ

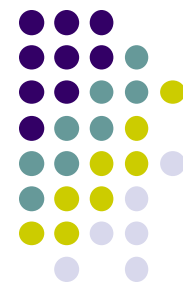


## ■ ການອະນຸຍາດ(Authorization)

- ການອະນຸຍາດເປັນການຖາມຄໍາຖາມກ່ຽວກັບສິດຜູ້ໃຊ້ລະບົບເພື່ອໃຫ້ຮູ້ໄດ້ຜູ້ໃຊ້ດັ່ງກ່າວມີສິດເຂົ້າເຖິງຂໍ້ມູນຢູ່ໃນລະບົບ ເຊັ່ນວ່າ: ຖາມຄໍາຖາມຜູ້ໃຊ້ວ່າ: “ເຈົ້າສາມາດເຮັດຫຍັງໄດ້ແດ່(what can you do)?” ຂະບວນການນີ້ຈະເປັນການຄວບຄຸມຊັບພະຍາກອນໃຫ້ກັບຜູ້ໃຊ້ທີ່ໄດ້ຮັບອະນຸຍາດເຂົ້າເຖິງຂໍ້ມູນໄດ້.
- ຊັບພະຍາກອນດັ່ງກ່າວລວມມີ ໄຟລຂໍ້ມູນ, ຖານຂໍ້ມູນ, ຕາຕະລາງ, ຂໍ້ມູນ, ການປະມວນຜົນຂໍ້ມູນ, ການຕັ້ງຄ່າໄຟລຂໍ້ມູນ ແລະ ການເຂົ້າເຖິງການເຮັດວຽກຂອງລະບົບຕ່າງໆ.



# ອົງປະກອບພື້ນຖານຂອງຄວາມປອດໄພ



## ■ ການກວດສອບ (Auditing)

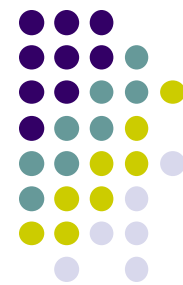
- ການກວດສອບເປັນການໃຫ້ລະຫັດ(Key) ໃຫ້ກັບຜູ້ໃຊ້ທີ່ຖືກອະນຸຍາດໃຫ້ເຂົ້າສູ່ລະບົບ, ເພື່ອຮັບປະກັນບໍ່ໃຫ້ຖືກປັດຕິເສດຈາກລະບົບ ແລະ ຜູ້ໃຊ້ດັ່ງກ່າວບໍ່ສາມາດໄປກຳນົດສິດທິບໍ່ອະນຸຍາດ(Deny)ໃຫ້ການເຮັດວຽກ ແລະ ການປະມວນຜົນຂໍ້ມູນ.
- ຕົວຢ່າງຢູ່ໃນລະບົບການຄ້າອີເລັກໂທຣນິກ(e-commerce system), ກົນໄກການກວດສອບແມ່ນມີຄວາມຈຳເປັນເພື່ອບໍ່ໃຫ້ຜູ້ໃຊ້ໄປກຳນົດສິດບໍ່ອະນຸຍາດ(Deny)ໃຫ້ມີການບໍລິການສັງຈອງສິນຄ້າ ຫຼື ລະລະບັງການບໍລິການ.

# ອົງປະກອບພື້ນຖານຂອງຄວາມປອດໄພ



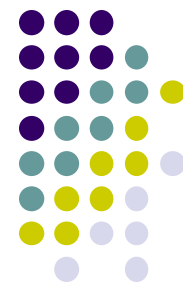
- ຄວາມລັບຂອງຂໍ້ມູນ (**Confidentiality**)
  - ຄວາມລັບ ຫຼື Privacy ເປັນຂະບວນການໃຫ້ກຳນົດໃຫ້ຂໍ້ມູນເປັນສ່ວນຕົວຈາກຜູ້ໃຊ້ອື່ນໆທີ່ບໍ່ກ່ຽວຂ້ອງ, ຂໍ້ມູນຈະຖືກເຜີຍແຜ່ ຫຼື ເຂົ້າເຖິງໄດ້ສະເພາະຜູ້ໃຊ້ທີ່ໄດ້ຮັບການອະນຸຍາດນັ້ນໄດ້.
  - ການເຂົ້າລະຫັດ(encryption) ຈະຖືກບັງຄັບໃຊ້ໃນການກຳນົດໃຫ້ສິດສ່ວນຕົວຂອງຂໍ້ມູນ.
  - ການກຳນົດສິດແບບ Access control lists (ACLs) ຈະບໍ່ກ່ຽວຂ້ອງກັບການບັງຄັບໃຊ້ສິດສ່ວນຕົວຂອງຂໍ້ມູນ.

# ອົງປະກອບພື້ນຖານຂອງຄວາມປອດໄພ

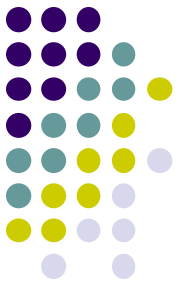


- ຄວາມຖືກຕ້ອງຂໍ້ມູນ(Integrity )
  - ຄວາມຖືກຕ້ອງຂໍ້ມູນເປັນການຮັບປະກັນວ່າຂໍ້ມູນມີຄວາມເຊື່ອຖືໄດ້ ແລະ ໄດ້ຖືກປ້ອງກັນການສູນເສຍ(accidental or deliberate or malicious)ທີ່ອາດຈະເກີດຂຶ້ນກັບຂໍ້ມູນເມື່ອມີການປ່ຽນແປງ ຫຼື ປັບປຸງຂໍ້ມູນ.
  - ຄ້າຍຄືກັບການໃຫ້ສິດສ່ວນຕົວ(Privacy), ການປ້ອງກັນຈະນຳໃຊ້ລະຫັດ(Key) ໃຫ້ກັບຂໍ້ມູນໃນຂະນະທີ່ສົ່ງຂໍ້ມູນຜ່ານເຄືອຂ່າຍໄປຍັງປາຍທາງ. ວິທີການກຳນົດລະຫັດອາດຈະນຳໃຊ້ຫຼັກການ hashing techniques ແລະ ການຢືນຢັນດ້ວຍລະຫັດ(message authentication codes)

# ອົງປະກອບພື້ນຖານຂອງຄວາມປອດໄພ



- ຄວາມພ້ອມໃຫ້ບໍລິການຂໍ້ມູນ(**Availability** )
  - ຈາກກແນວຄວາມຄິດຄວາມປອດໄພ, ຄວາມພ້ອມໃຫ້ບໍລິການຂໍ້ມູນໝາຍເຖິງການຮັບປະກັນບໍລິການຂໍ້ມູນໃຫ້ກັບຜູ້ໃຊ້ທີ່ໄດ້ຮັບອະນຸຍາດເຂົ້າໃຊ້ຂໍ້ມູນໄດ້ເມື່ອຕ້ອງການ.
  - ຄວາມພ້ອມໃຫ້ບໍລິການຂໍ້ມູນຈະເປັນເປົ້າໝາຍຂອງຜູ້ບຸກໂຈມຕີລະບົບ, ຊຶ່ງຈະພະຍາຍາມປົດການບໍລິການ ແລະ ສະກັດກັ້ນບໍ່ໃຫ້ຜູ້ໃຊ້ສາມາດນຳໃຊ້ຂໍ້ມູນນັ້ນໄດ້.



# ໄພຄຸກຄາມ

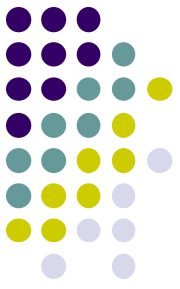
## ■ ໄພຄຸກຄາມ(Threats)

- ໄພຄຸກຄາມໝາຍເຖິງທຸກບັນຫາທີ່ກໍ່ໃຫ້ເກີດຄວາມເສຍຫາຍຂຶ້ນກັບຂໍ້ມູນ ຫຼື ຊັບສິນ ແລະ ຕໍ່ອົງປະກອບຂອງຄວາມປອດໄພດ້ານໃດດ້ານໜຶ່ງ.
- ໄພຄຸກຄາມອາດຈະບໍ່ເກີດຂຶ້ນກໍ່ເປັນໄປໄດ້ຖ້າມີການປ້ອງກັນທີ່ດີ ຫຼື ມີວິທີຈັດການກັບໄພຄຸກຄາມ, ການກະທຳທີ່ກໍ່ໃຫ້ເກີດການເສຍຫາຍຂອງຂໍ້ມູນເອີ້ນວ່າ: ການບຸກໂຈມຕີ(Attack), ຜູ້ທີ່ກະທຳ ຫຼື ຜູ້ທີ່ເປັນເຫດກໍ່ໃຫ້ເກີດເຫດການດັ່ງກ່າວເອີ້ນວ່າ: ຜູ້ບຸກໂຈມຕີ (Attacker)



# ຊ່ອງໂວ່(Vulnerability)

- ຈຸດອ່ອນ ຫຼື ຊ່ອງໂວ່(Vulnerability)
  - ຈຸດອ່ອນ ຫຼື ຊ່ອງໂວ່ໝາຍເຖິງຂໍ້ຜິດພາດຂອງລະບົບທີ່ ພາໃຫ້ເກີດມີຄວາມສ່ຽງຕໍ່ການຄຸກຄາມ.
  - ຈຸດອ່ອນອາດຈະເກີດຈາກການອອກແບບລະບົບພິດພາດ(Poor design), ການຕັ້ງຄ່າລະບົບຜິດພາ (configuration mistakes), ຫຼື ຜິດພາດທາງດ້ານ ເຕັກນິກການຂອງໂປຣແກຣມ ເຊັ່ນວ່າ: ການກວດສອບ ການປ້ອນຂໍ້ມູນຂອງຜູ້ໃຊ້(Weak input validation) ເຮັດໃຫ້ເກີດຊ່ອງໂວ່ຕໍ່ການບຸກໂຈມຕີ ດ້ວຍວິທີການ ປ້ອນຂໍ້ມູນ.



# ການບຸກໂຈມຕີ(Attacks)

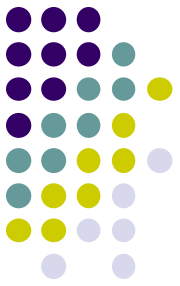
- ການບຸກໂຈມຕີ(Attacks)
  - ການບຸກໂຈມຕີເປັນເຫດການໜຶ່ງທີ່ດໍາເນີນການຜ່ານຈຸດອ່ອນ ຫຼື ຊ່ອງໄວ່ ຫຼື ຄວາມສ່ຽງຂອງລະບົບ ເຊັ່ນວ່າ: ການບຸກໂຈມຕີດ້ວຍການສົ່ງຄໍາສັ່ງ(malicious input) ຜ່ານໂປຣແກຣມ ຫຼື ບຸກໂຈມຕີເຄືອຂ່າຍ (flooding a network) ເພື່ອຢຸດການບໍລິການຂອງລະບົບ.



# ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ

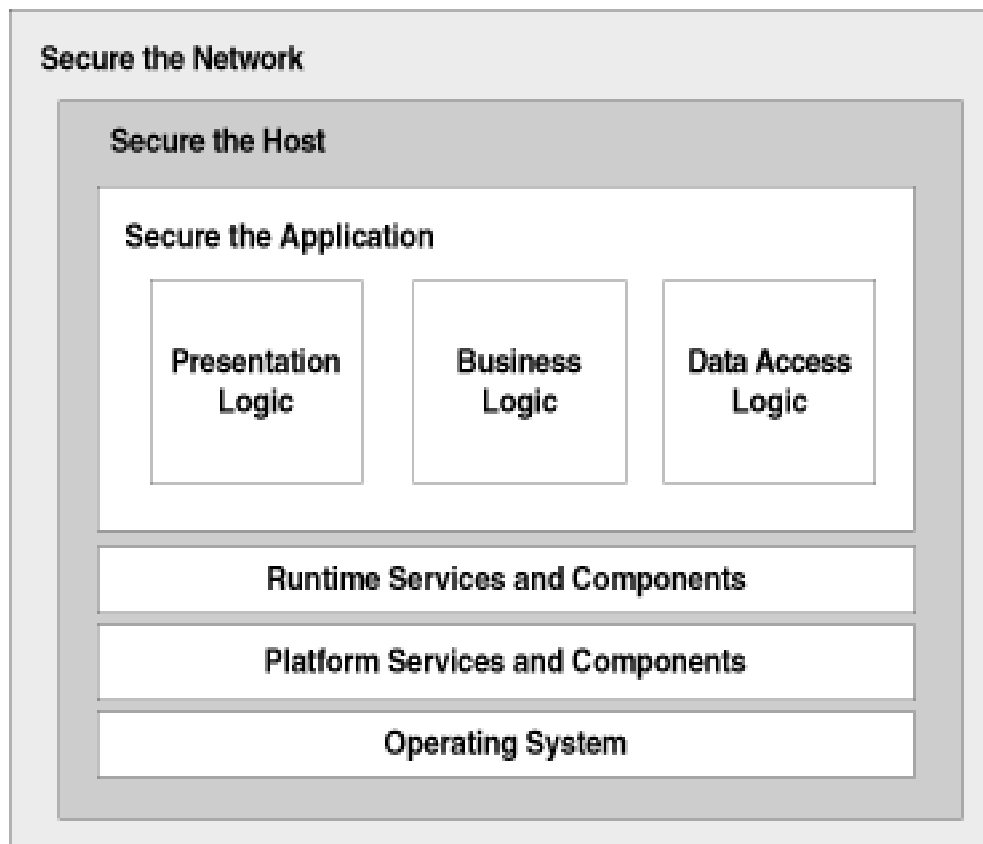
- ການອອກແບບ ແລະ ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ ຈຳເປັນຕ້ອງຮູ້ຈັກຄວາມສ່ຽງທີ່ອາດຈະເກີດຂຶ້ນ. ແບບຈຳລອງຄວາມສ່ຽງ(Threat Modeling) ເປັນສ່ວນປະກອບສຳຄັນໃນການອອກແບບເວບ, ເພື່ອວິເຄາະສະຖາປັດຕະຍະກຳຂອງເວບ, ອອກແບບ ແລະ ຮູ້ໄດ້ຈຸດອ່ອນ ຫຼື ຊ່ອງໄວຂອງໂປຣແກຣມ.
- ການພັດທະນາໂປຣແກຣມຕ້ອງໃຫ້ຕອບສະໜອງຄື: ເຄືອຂ່າຍປອດໄພ(secure network), ຄອມພິວເຕີປອດໄພ(secure host) ແລະ ການຕັ້ງຄ່າໄຟລໂປຣແກຣມປອດໄພ(secure application configuration)

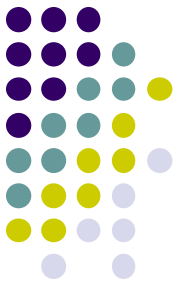




# ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ

- ລຳດັບຊັ້ນຂອງຄວາມປອດໄພ





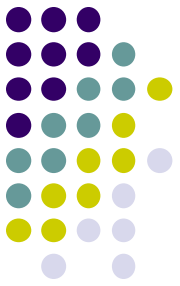
# ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ

- ເຄືອຂ່າຍມີຄວາມປອດໄພ
  - ປະເພດຄວາມປອດໄພຂອງເຄືອຂ່າຍຕ້ອງປະກອບມີອຸປະກອນຄື:
    - routers
    - Firewalls
    - switches.
  - ການກຳນົດສິດທິ(Role) ໃຫ້ຜູ້ໃຊ້ເຂົ້າເຖິງຂໍ້ມູນຢ່າງມີຄວາມປອດໄພ, ມີການຈັດການປ້ອງກັນການບຸກໂຈມຕີຜ່ານທາງເຄືອຂ່າຍ.

# ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ



- ຄອມພິວເຕີມີຄວາມປອດໄພ
  - ຄວາມປອດໄພຂອງຄອມພິວເຕີປະກອບມີໂປຣແກຣມ ຫຼື ບໍລິການຕ່າງໆໃນເຄື່ອງເຊີເວີຕ້ອງມີການຕັ້ງຄ່າໃຫ້ມີຄວາມປອດໄພ ເຊັ່ນວ່າ: ເວບເຊີເວີ(Web server), ໂປຣແກຣມ (Application server), ຖານຂໍ້ມູນ(Database Server).
  - ເມື່ອມີການຕິດຕັ້ງໂປຣແກຣມໃໝ່ລັງໃນເຄື່ອງຄວນພິຈາລະນາວ່າມີຜົນກະທົບຕໍ່ຄວາມປອດໄພບໍ່ ເຊັ່ນວ່າ: ໂປຣແກຣມມີການສ້າງ Account ບໍ່? ມີການເພີ່ມບໍລິການອັດຕະໂນມັດບູ? ຜູ້ໃຊ້ຄົນໃດທີ່ໃຊ້ບໍລິການດັ່ງກ່າວ? ມີການສ້າງ Script ຂຶ້ນມາໃໝ່ບໍ່?



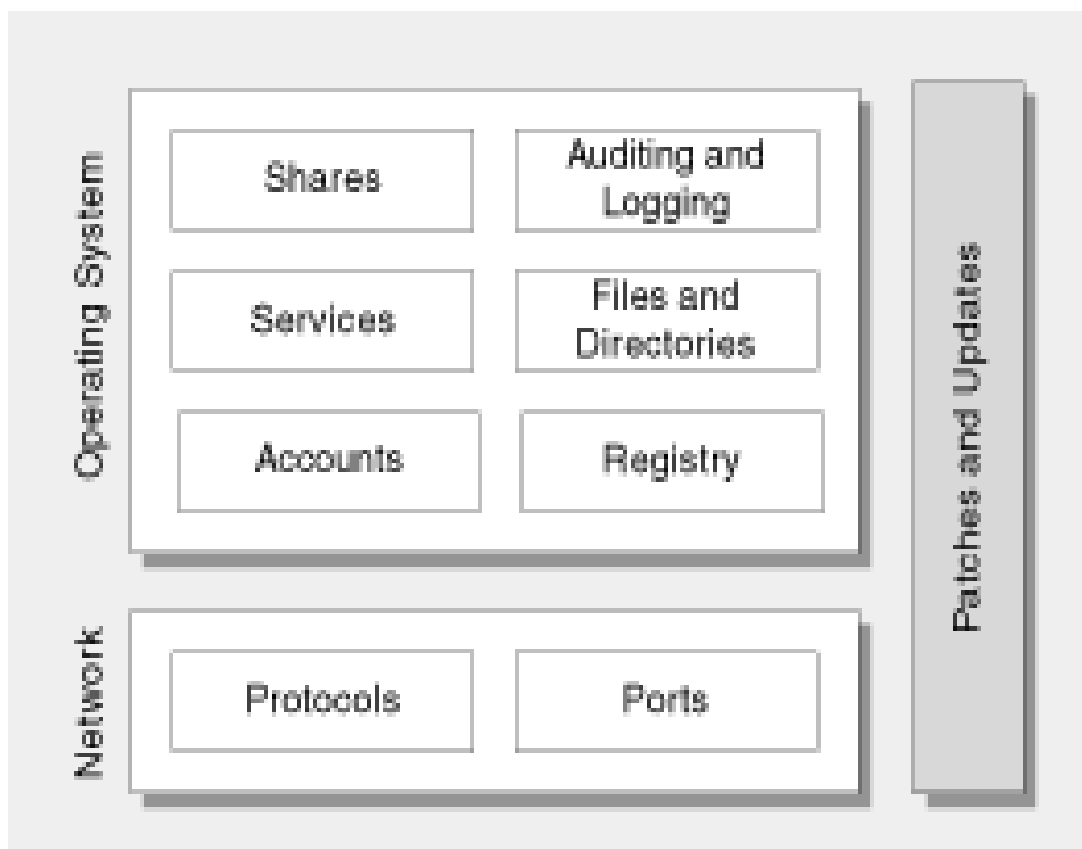
# ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ

- ປະເພດຈຸດອ່ອນຂອງໂປຣແກຣມ
  - Patches and Updates
  - Services
  - Protocols
  - Accounts
  - Files and Directories
  - Shares
  - Ports
  - Auditing and Logging
  - Registry

# ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ



- ຄອມພິວເຕີມີຄວາມປອດໄພ

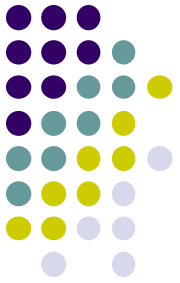




# ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ

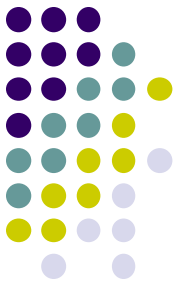
- ໂປຣແກຣມມີຄວາມປອດໄພ
  - ໂປຣແກຣມທີ່ມີຄວາມປອດໄພຕ້ອງມີການອອກແບບ ແລະ ສ້າງຂຶ້ນໂດຍຜ່ານການວິເຄາະຮູ້ໄດ້ເຖິງຄວາມສ່ຽງ, ຈຳແນກໄດ້ຄວາມສ່ຽງແຕ່ລະໆດັບ, ຮູ້ໄດ້ຈຸດອ່ອນ ແລະ ວາງແຜນບໍລິຫານຈັດການກັບຄວາມສ່ຽງດັ່ງກ່າວໃຫ້ສາມາດຄວບຄຸມໄດ້.
  - ຖ້າໂປຣແກຣມບໍ່ມີການກວດສອບການປ້ອນຂໍ້ມູນຂອງຜູ້(input validation) ເຮັດໃຫ້ເກີດມີຄວາມສ່ຽງ ແລະ ເປັນຊ່ອງໄວ້ໃຫ້ແກ່ການບຸກໂຈມຕີຜ່ານທາງໂປຣແກຣມ ເຊັ່ນວ່າ: SQL Injection, Cross site scripting ແລະ ອື່ນໆ.

# ສ້າງເວບໃຫ້ມີຄວາມປອດໄພ



- ປະເພດຈຸດອ່ອນຂອງໂປຣແກຣມ
  - Input Validation
  - Authentication
  - Authorization
  - Configuration Management
  - Sensitive Data
  - Session Management
  - Cryptography
  - Parameter Manipulation
  - Exception Management
  - Auditing and Logging

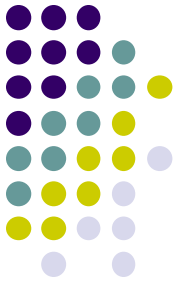
# ສະຫຼຸບ



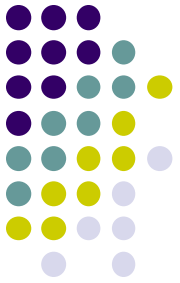
- ຄວາມປອດໄພຂອງ Firewall ແລະ ເຄື່ອງເຊີເວີບໍ່ສາມາດປ້ອງກັນການບຸກໂຈມຕີໄດ້ໝົດ.
- ຄວາມປອດໄພຂອງໂປຣແກຣມຕ້ອງປ້ອງກັນຢູ່ໃນ 3 ລະດັບຄວາມປອດໄພ ຄື: ລະດັບເຄືອຂ່າຍ(network layer), ລະດັບເຄື່ອງເຊີເວີ(host layer), ລະດັບໂປຣແກຣມ (application layer).
- ນອກຈາກນັ້ນ, ການອອກແບບ ແລະ ສ້າງໂປຣແກຣມກໍຕ້ອງນຳໃຊ້ການອອກແບບຄວາມປອດໄພພ້ອມ ແລະ ພັດທະນາຕາມຍຸດທະສາດຄວາມປອດໄພ (security principles).



# ຂໍ້ມູນອ້າງອີງ



- [1] J.D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla and Anandha Murukan. Improving Web Application Security, Microsoft Corporation, 2013
- [2] ການປ້ອງກັນ ແລະ ຮັກສາຄວາມປອດໄພເຄືອຂ່າຍ,  
[www.mict4u.net](http://www.mict4u.net)



ព្យាបាល និង ព័ត៌មាន

ឧបករណ៍