Komplex alkalmazási környezetek felderítése és menedzsmentje

(Mérési feladatok)

Szatmári Zoltán, Izsó Benedek
Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

2015. szeptember 10.

1. Bevezető

A mérés során a segédletben megismert eszközöket kell alkalmazni a beállítások ellenőrzésére, illetve a nem ismert részek felderítésére. Így az első blokk egy általunk nem ismert infrastruktúra felderítéséről fog szólni, aminek végén a szöveges SMTP protokollt is ki kell próbálni. A második blokkban egy webszolgáltatást nyújtó komplex alkalmazási környezet megismerése lesz a cél, aminek különböző konfigurációit ki kell egészíteni, vagy éppen egy hiba következményét észlelve szisztematikusan meg kell keresni a hiba okát, és javítani magát a hibát (így megszüntetve ezt a hiba hatásláncot). A harmadik blokk a működő rendszer monitorozásáról, a monitorozó rendszer beállításáról, kiegészítéséről fog szólni.

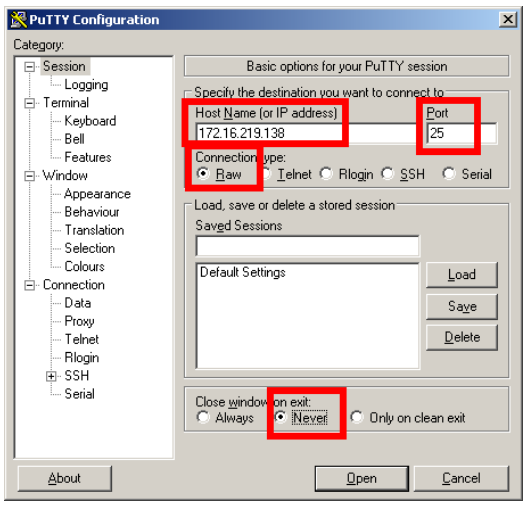
A gyakorlatban elsajátítandó módszerek, eszközök saját alkalmazásunk fejlesztésénél is előjönnek. Hálózatot használó, több komponensből álló alkalmazások esetén azok környezetét helyesen és biztonságosan be kell állítani, illetve fel kell készíteni saját programunkat arra, hogy a menedzsment csapat annak állapotát monitorozni tudja. Hiba esetén fontos, hogy az üzemeltető akár SMS riasztást kaphasson, és azonnali beavatkozással javíthassa a hibát.

1.1. A mérési infrastruktúra

A mérésen 4 darab virtuális gép áll rendelkezésre. Az egyik egy Windows 7 operációs rendszerrel felszerelt gép, ami a kliens szerepét tölti be. A többi (VM1-3), Ubuntu Linux 12.04 LTS operációs rendszert futtató gép, amik szolgáltatásokat nyújtanak. Ezeket kell majd először felderíteni, majd helyesen bekonfigurálni, végül pedig monitorozni.

A gépekre a feladatok elvégzéséhez szükséges eszközök előre telepítve vannak, így a mérés során csupán ezek megfelelő felhasználása a feladat. A kliens gép fontosabb mérést támogató szoftverei és **jó tanácsok a méréshez**:

- Putty: SSH és Telnet kliens. Lehetővé teszi távoli géphez való kapcsolódást titkosított csatornán (SSH, 22-es port), titkosítatlan csatornán (Telnet, 23-as port), vagy bármilyen szöveges protokollt használó programhoz (Raw üzemmód). SMTP és HTTP kiszolgálókhoz való kapcsolódást is ***Raw*** üzemmódban lehet megtenni, ekkor be kell állítani a ***portot***, és a *Close window on exit* opciói közül is érdemes a ***Never*** beállítást választani, hogy a kimenet a lefutás után is olvasható maradjon. Egy ilyen beállítást mutat a lenti ábra:



- WinSCP: fájlok titkosított másolását teszi lehetővé minden gépre (vagy gépről), amely futtatja az ssh daemont
- Firefox: böngésző
- Zenmap: grafikus Nmap frontend
- Wireshark: hálózati forgalmat elemző eszköz

A mérés során célszerű az egyes feladatpontok szövegét figyelmesen végigolvasni és csak utána végrehajtani, különben fontos információk kerülhetik el a figyelmünket. Javasolt a kliens gépen dolgozni, ahonnan a többi kiszolgáló SSH kapcsolaton keresztül elérhető. Többször szükséges lesz egy-egy konfigurációs állomány minimális módosítása. Ha a keresett opciót nem találjuk benne, akkor *bátorkodjunk segítséget keresni az Interneten*! Előfordulhat, hogy egy részfeladat „megoldása” után mégsem működik a teljes rendszer. Ez a feladat része! Folytassuk a hibakeresést, derítsük ki az okot a tanult eszközök alkalmazásával, végül pedig javítsuk és dokumentáljuk!

1.2. A mérés számonkérése

A mérésre kapott jegy *szóbeli beszámoló* eredményeként fog előállni. Magát a mérést párosan kell elvégezni, és a beszedés is részben párosan fog zajlani, de minden hallgató külön-külön osztályzatot kap, így senkinél sem lehetnek ismeretlen részek. A beszámolón három feladattípus jöhet elő: mérési útmutatóból (illetve hivatkozásaiból) megismerhető elméleti tudás visszakérdezése, jegyzőkönyv feladatainak bemutatása, helyszínen megoldandó gyakorlati példa. Az alábbiakban ezen feladattípusokra látunk néhány példát.

1. Egy mérési útmutatóban megismert technológia vagy módszer bemutatása, például:

   - Tudni kell számolni az alhálózatokkal, IP címekkel, alhálózati maszkokkal. A CIDR jelölést és broadcast cím jelentését, annak tipikus értékét ismerni kell.
   - El kell tudni mondani a DNS névfeloldás és a DNS lekérdezés folyamatát, illetve tudni kell legalább két lekérdező eszköz nevét.
   - Hogyan szűr a tűzfal? Miből épül fel a szabály? Az hogyan rendeződik a szűrés elvégzéséhez? Mi történik, ha egy csomagra egy szabály sem alkalmazható?
   - Mondj legalább három fajta letapogatási módszert (amit az nmap is alkalmaz), amivel a távoli gépen lévő nyitott portok kideríthetőek! Mi az OS fingerprinting lényege?
   - Mire jó, és hogyan működik a virtualhosting?
   - Hogy néz ki a Nagios architektúrája? Hogyan tudja a Nagios távoli gép merevlemez használatát monitorozni?
   - Mi a Nagios command (vagy más néven plugin)? Mire használható? Mi az, amit a Nagios rendszerhez csatlakozáshoz tudnia kell?

2. A jegyzőkönyv általunk kért részének bemutatása, elmagyarázása.

3. Egy helyszínen megoldott feladat, melyre a mérés megoldásával lehet felkészülni. Többek között:

   - Tudni kell a tűzfalat kezelni, tisztában kell lenni azzal, hogy mit jelent, hogy egy adott interfészen, illetve adott porton hallgatózik egy alkalmazás.
   - Tudni kell szöveges protokollt interaktívan lejátszani, illetve azt Wireshark eszközzel elemezni.

- Virtualhost-ot tudni kell konfigurálni.
- Nagios plugint tudni kell írni, és rendszerbe helyezni.

Természetesen a konfigurációk (pl. Apache, Nagios), programozási nyelvek (pl. bash) bemagolása nem követelmény, de példák, manual-ok, referenciák alapján meg kell tudni oldani a feladatot!

2. Mérési feladatok

A következő feladatokat kell mérőpárban megoldani és annak menetéről jegyzőkönyvet készíteni.

2.1. Infrastruktúra elindítása

A mérésen 4 darab virtuális gép áll rendelkezésre, mely virtuális gépeket a VCL felhőben lehet elindítani. A VCL használatáról bővebb információ az online oktatási anyagok között érhető el[1]. A VPN kapcsolat létrehozása után BME Címtáras azonosítást használva egy ITLab2-MIT1 környezetet kell foglalni, ami mind a 4 virtuális gépet elindítja. Arra ügyeljünk, hogy a **lefoglalt időtartam letelte után a virtuális gépek leállnak, és minden módosítás elveszik**! Emellett ha 15 percig nincs bejelentkezés, akkor is megszakad a kapcsolat. Ekkor újat kell foglalni (ami új, alapállapotban lévő virtuális gépeket indít), a régi foglalást pedig a „Remove" gombbal fel kell szabadítani! Ezért a jegyzőkönyvet készítsük saját gépen, így ha a virtuális gépeket újra kell indítani, akkor is gyors konfigurálás után az utolsó állapotba térhetünk vissza.

Mind a Linuxos, mind a Windows-os környezetbe a *VCL felületen megjelenő felhasználónévvel* kell távolról bejelentkezni. Ha ez a felhasználó nincs bejelentkezve egyik gépre sem, 15 perc időtúllépés után megszakad a kapcsolat, és újra kell foglalni. Windows-ba bejelentkezve alapból rendszergazda hozzáférés fogad. Linuxba a megfelelő környezet eléréséhez érdemes a következőképpen bejelentkezni:

- putty programmal SSH-n keresztül belépni a *VCL által megadott bejelentkezési adatokkal*
- `su meres` parancs kiadása után `LaborImage` jelszóval lehet átjelentkezni egy olyan felhasználóra, akinek már helyesek a környezeti beállításai

[1] https://www.inf.mit.bme.hu/wiki/it/szolgaltatasok/cloud

- végül pedig `sudo bash` paranccsal lehet root jogosultságot szerezni, szintén `LaborImage` jelszóval

Ügyeljünk arra, hogy a kliensek UTF-8 kódolást használnak, amit puttyban a Windows | Translation kategória legördülő menüjében tudunk igazítani. (Enélkül többek között a Midnight Commander vonalai helyett „ä" betűk jelennének meg.)

2.2. Infrastruktúra felderítése

Idegen gépek szkennelése éles infrastruktúrán nem megengedett, hálózat elleni támadásnak minősül, kitiltást vonhat maga után.
A mérés során mindenki a saját infrastruktúrájával foglalkozzon, más hallgatók mérését ne zavarja! A szándékos piszkálás jegylevonást von maga után, a virtuális laboron kívüli (nem 10-zel kezdődő IP című) gépek szkennelése pedig TILOS.

Az elindított infrastruktúráról sajnos eddig kevés információval rendelkezünk, így a feladat első felében ezt kell lépésről-lépésre felderíteni.

1. A mérés során az infrastruktúra modelljét folyamatosan kell készíteni, ahogy egyre több információ elérhetővé válik. Készítsük el az infrastruktúra ábra első változatát, amin az IP címek már szerepelnek. Ezen információkra szükség lesz a mérés során.

2. A VCL által megadott adatokkal lépjünk be a (Windowsos) kliens számítógépre. Jelenleg *melyik alhálózathoz* csatlakozunk? Írd fel CIDR jelöléssel! Vegyük észre, hogy a 10.82-vel kezdődő IP cím az a cloud management hálózatához tartozik, azzal felhasználóként nem tudunk mit kezdeni. A mérés során használjuk a másik interfészt!

   Tegyük fel, hogy egy gép szkenneléséhez 1 másodperc kell. Mennyi idő alatt lehetne az egész hálózatot végignézni?

3. A VCL-ből megtudható a 3 Linuxos kiszolgáló IP címe is. Ezen három IP cím által meghatározott *legszűkebb alhálót* számítsuk ki! Mi lett az eredmény?

4. Végezzük el a megállapított alhálón az infrastruktúra felderítését a Zenmap eszközzel a „Ping Scan" eljárást használva! (Ha az alhálóban az 1-esek száma 23 vagy kevesebb, akkor szkenneljük a /23-as alhálót,

mert több gép szkennelése sok időt venne igénybe.) Milyen információkat gyűjthetünk ezzel a módszerrel? Egészítsük ki az infrastruktúra ábrát a hozzánk tartozó három Linuxos VM-ekről kiderített új információkkal!

5. Futtassunk a „Quick scan" vizsgálatot a VCL-ből megismert három Linuxos gépre! Milyen új adatokat fedezhetünk fel? Jelöljük be a készülő infrastruktúra ábrán! (Célszerű lehet a Zenmap programot újraindítani, ezáltal tiszta környezetet kapunk benne.)

6. Futtassunk egy újabb felderítést „Intense Scan" eljárást használva, most is csak a három darab saját, szolgáltatásokat nyújtó gépre. Egy újabb diagram formájában készítsük el az infrastruktúra modelljét, vegyük fel a friss paramétereket az ábrához! Mik az új információk?

7. Amennyiben már minden külső forrásból elérhető információ rendelkezésünkre áll, akkor a VCL-ből kiolvasható adatokkal (IP cím, felhasználónév, jelszó) lépjünk be a szolgáltatásokat nyújtó Linuxos virtuális gépekre, és támasszunk alá legalább három eddigi megállapítást a gépeken elérhető tényleges információkkal.

8. A megismert információkat felhasználva a mérési segédletben bemutatott módszerrel küldjünk e-mailt! A VM1 gép rendelkezik SMTP szolgáltatással a mail.konyv.hu tartományra. Küldjünk a ***Windowsos kliensről*** a `putty` program segítségével elektronikus levelet a `root@mail.konyv.hu` címre! A levél megérkezését ellenőrizhetjük a VM1 gépen a `/var/mail/username` fájl tartalma alapján, ahol alapértelmezetten a *root* felhasználó levelei tárolódnak.

   A levelezőszerver SPAM védelem miatt kis késleltetéssel jelentkezik be, így nehezítve az „Early speaker"-nek nevezett sietős spammerek életét. Várjunk türelemmel! A levél tartalmi részében (a DATA kulcsszó után) ne foglalkozzunk a fejléccel, csak valami gyors, rövid tartalmat küldjünk. Az esetleges hibákat hárítsuk el!

   Bónusz feladat: Mi történik, ha a kliensről a `root@konyv.hu` címre küldjük a levelet (a mail aldomaint nem használva)? Mit jelent az SMTP üzenet?

2.3. Webszolgáltatás

A megismert infrastruktúrában a webkiszolgálás jelentős szerepet játszik, de a jelenlegi szolgáltatás beállításaiban hibák lehetnek. Feladatunk egy egyszerű PHP webalkalmazás, majd egy WordPress blog üzembe helyezése. A

művelet során a megismert diagnosztikai módszerekkel *állapítsuk meg a hibák okait és hárítsuk el azokat.*

9. Teszteljük a VM1 gép webkiszolgálását a kliens gépről egy webböngészővel. A szerver IP címe alapján kérjük le az ott kiszolgált weboldalt. Türelmesen várjuk meg a végeredményt! Mit tapasztalunk?

10. Korábbi ismereteinket felhasználva fogalmazzuk meg mi a különbség a kapott hibaüzenet és a HTTP 404-es hibaüzenet között?

11. A Wireshark eszközzel vizsgáljuk meg, milyen hálózati forgalmat tapasztalunk. Állítsuk be úgy az eszközt, hogy csak a webkiszolgáláshoz kapcsolódó csomagokat mutassa. Mi a filter kifejezés? Határozzuk meg, hogy mi lehet az előbb tapasztalt hiba oka és hárítsuk azt el. Indokoljuk döntésünket!

12. A webszolgáltatás működését kliens oldalon különböző módszerekkel tudjuk ellenőrizni. Soroljunk fel ezek közül néhányat, és ne feledkezzünk meg arról sem, hogy nem minden kliens rendelkezik grafikus felülettel! Egy konzolos módszert próbáljunk is ki!

13. Vizsgáljuk meg a webszerver beállításait és állapítsuk meg, milyen virtuális kiszolgálókat (virtualhost) üzemeltet!

14. Tekintsük meg a kliens böngészőjében az összes ilyen oldalt! Mit tapasztalunk? Keressünk megoldást a problémára és tegyük meg a szükséges beállításokat.

15. Az így már működőképes webkiszolgálás hálózati forgalmát vizsgáljuk meg a Wireshark eszközzel és mutassuk be a hálózati forgalom elemzése alapján a *virtualhosting* működési elvét. Figyeljünk a cache-re is! (Firefox-ban Beállítások | Adatvédelem | törölni az előzményeket link.)

16. A konyv.hu domain nevet sajnos még nem jegyezte be a hatóság, de már most, a bejegyzés előtt szeretnénk webalkalmazást fejleszteni. A VM1 gép *hosts* fájlában állítsuk be helyesen az adatbázisszerver `sql.konyv.hu` hoszthoz tartozó IP címet.

17. Ellenőrizzük egy előre telepített webalkalmazás működését és töltsük be a böngészőbe az alapértelmezett (IP címen elérhető) virtualhost alatt elérhető `konyvek.php` oldalt. Az esetleges lassabb reakciót türelemmel várjuk meg. Mi a hibajelenség oka? Oldjuk fel a problémát! Kiinduláshoz segítséget találhatunk az alapértelmezett virtualhost phpinfo.php oldalán. (Figyeljünk arra, hogy a probléma többrétű is lehet, nem biztos, hogy egy opció módosításával megoldható.)

18. A webalkalmazás kódjában vizsgálódva megtalálhatjuk az adatbázis csatlakozáshoz használt felhasználót és jelszót. A VM1 géptől a kliensről lekérve a `http://<vm1.ip.cim>/phpmyadmin/` URL-el elérhető tartalmat, a népszerű webes adatbázis menedzsment felülethez jutunk. Az adatbázisban a `konyv` táblába helyezzünk el néhány példa adatot, majd vizsgáljuk meg, hogy a webalkalmazásban is megjelent-e! Mi volt a jelszó? Hogyan sikerült megjeleníteni?

19. Telepítsük fel az infrastruktúránkra a WordPress blogmotort. A működéséhez hozzunk létre egy wp.konyv.hu virtualhostot, és a motorhoz külön adatbázist saját felhasználóval. Az adatbázis manipulálását (például új felhasználó, tábla felvételét) a phpmyadminnal lehet megtenni. Az ehhez szükséges *root* jelszót a mostani feladattól kezdve lehet használni, ez pedig LaborImage. A virtualhostot az előzőek mintájára az Apache konfigurációs mappájában a `sites-available` mappa alatt hozzuk létre, majd az `a2ensite vhost.neve.hu` utasítással engedélyezzük. A WordPress blogmotor telepítéséhez segítséget találunk az Interneten vagy a letöltött csomagban is. A sikeres telepítés után publikáljunk egy bejegyzést, demonstrálva a helyes működést. Készítsünk róla képernyőképet is a jegyzőkönyvbe. (A WordPress motort a http://wordpress.org/download/ címről érdemes leszedni saját gépre, utána pedig WinSCP segítségével lehet a Linuxos virtuális gépekre másolni!)

2.4. Rendszermonitorozás

A megismert szolgáltatások egy részéhez előre beállított monitoring rendszer üzemel, melyet szeretnénk kiterjeszteni a webes szolgáltatásra is. A következő feladatok során vizsgáljuk meg a monitoring rendszer működését és a felmerülő hibákat hárítsuk el!

20. A kliens gépen tekintsük meg a Nagios rendszer webes felületét (http://<*vm3.ip.cim*>/nagios3), bejelentkezni a nagiosadmin/LaborImage adatok megadásával lehet. Tekintsük meg a *Services* oldalt. Vizsgáljuk meg és vessük össze a topológia ábránkkal, hogy milyen szolgáltatásokat monitorozunk a jelenlegi állapotban.

21. Vizsgáljuk meg a szerver oldalon a központi Nagios monitoring rendszer konfigurációit és a hosztok beállításainál javítsuk ki az IP cím beállításokat! A beállítások módosítása és a monitoring szerver újraindítása után vizsgáljuk meg a webes felületen, hogy helyesen működik-e a monitoring funkció! (Ha valami még nem stimmel, javítsuk!)

Az állapotlekérdezések időzítve vannak. Azok hamarabbi elvégzését a szolgáltatásokat listázó oldalon (*Services*) egy hosztot kiválasztva lehet ütemezni, a *Scedule a check of all services on this host* opciót kiválasztva. Milyen hibát tapasztalunk? Hárítsuk el a problémát!

22. A webszerveren fut már az NRPE ágens, de a központi monitoring rendszer konfigurációs beállításai nincsenek felkészítve a webszerver monitorozására. Vegyük fel a webszervert új hosztként és állítsuk be rajta, hogy az alapvető NRPE-n keresztül elérhető szolgáltatásokat monitorozzuk, illetve a PING alapú elérhetőséget. (Ne felejtsük el a Nagios szerveralkalmazást újraindítani!)

23. Tekintsük meg a Nagios webes felületén az előző lépésben tett módosítások eredményét. Keressük meg a hiba okát és javítsuk ki!

24. A webszerveren definiáljunk saját monitoring feladatot, mely azt vizsgálja, hogy a korábban telepített WordPress blog wp-settings.php állományát tudja-e valaki írni. Ha igen, akkor a „CRITICAL” riasztást kell adni, különben „OK” legyen a szolgáltatás állapota. Módosítsuk az NRPE és a központi monitoring rendszer beállításait, hogy ez a paraméter is megjelenjen a webes felületen.

2.5. Mérés értékelése

25. Mit tanultál a mérésből?

26. Mely részeket volt könnyű megoldani?

27. Mi okozott sok fejtörést?

28. Mely részek tetszettek, illetve mi állt távol tőled? Mi az amivel még mélyebben megismerkedtél volna?