Hálózati biztonság

2008

Tartalomjegyzék

1. Bevezetés

Egy számítógépes hálózat védelme a következő alapvető intézkedésből áll: a rendszergazda által felépített biztonsági házirend, amely a hálózatot és a hálózati erőforrásokat a jogosulatlan hozzáféréstől, használattól, változtatástól hivatott megvédeni.

Összehasonlítás a számítógép védelmével

Egy hálózat védelme hasonlít egy ország lehetséges támadási pontjainál alkalmazott megfelelő védelmi eszközök telepítéséhez. A számítógépes védelem sokkal többet jelent, mint egyetlen PC védelme a külső támadások ellen. Megelőző lépésekkel próbálják megvédeni az egyes számítógépeket, magát a hálózatot, azáltal, hogy védik a számítógépeket és egyéb megosztott erőforrásokat, mint például a nyomtatók, és a hálózatra kapcsolt háttértárolók. A támadásokat meg lehet állítani a behatolási pontoknál, mielőtt még elterjedhetnének. Ezzel ellentétben a számítógépes védelem az egyes számítógépes hosztok védelmére koncentrál, és ennek érdekében tesz lépéseket. Egy számítógépes hoszt, melynek a védelme veszélyeztetett, feltehetőleg megfertőzhet más hosztokat, melyek egy potenciális nem védett hálózathoz kapcsolódnak. A magasabb hozzáférési prioritással rendelkező felhasználók nagyobb sebezhetőséget jelentenek a számítógépes hosztvédelmet tekintve, mivel a kártékony programok zöme a felhasználói jogosultságokat képes használni.

Egy védett hálózat részei

A hálózati biztonság a felhasználók azonosításával kezdődik, például felhasználónévvel és jelszóval. Az azonosítás után egy alapos tűzfal a megadott szabályok alapján engedélyezi, hogy mely erőforrásokhoz férhetnek hozzá a felhasználók. Ez hatásos a jogosulatlan hozzáférések megakadályozására, de nem tudja ellenőrizni a potenciális veszélyforrásokat, mint például a hálózaton továbbított férgeket. Egy IPS[1] segít a potenciális veszélyforrások felkutatásában és elhárításában, mint amilyen a malware[2]. Az IPS alkalmas a hálózat gyanús

[1] Intrusion Prevention System, behatolás-megelőző rendszer

[2] kártékony programok

forgalmának monitorozására, az anomáliák elleni védelemre, mint amilyen a DoS[1] támadás. A hálózaton folyó két hoszt közötti kommunikációt titkosítással priváttá tehetjük.

A *honeypot*[2] (szó szerint mézesbödön) alapvetően olyan hálózati csapda, melyeket a hálózatba lehet telepíteni előrejelző, felügyeleti eszközként. A támadók által használt technikákat, melyekkel veszélyeztetni próbálják az erőforrásokat, megfigyelik a támadás alatt és után, hogy megismerjék az új kizsákmányolási technikákat. Az ilyen analízisek elősegíthetik a biztonság fejlesztését, egy honeypot-tal védett hálózatban.

Biztonsági intézkedések:

A hálózatok biztonsági intézkedései minden szituációban mások. Egy otthoni vagy irodai hálózatnak csak alapvető biztonságra van szüksége, míg egy nagyvállalatnak magas szintű és fejlett szoftverre és hardverre van szüksége, hogy megakadályozza a szándékos támadásokat, mint például a hacker és spam[3] támadások.

Otthoni hálózatok:

- Alap tűzfal.
- Windows felhasználóknak alapvető antivírus szoftver és tűzfal, mint a McAfee, a Norton AntiVirus, a NOD, az AVG Antivirus, vagy a Windows Defender, de más programok is megfelelőek, ha tartalmaznak egy víruskeresőt a kártékony szoftverek felkutatására.
- Vezeték nélküli kapcsolat estén robosztus jelszó használata.

Közepes méretű vállalatok:

- Egy meglehetősen erős tűzfal.
- Egy erős antivírus- és internetvédelmi szoftver.

---

[1] Denial of Service – szolgáltatásmegtagadásos támadás

[2] Honeypot szó szerint mézescsuprot jelent. A mézescsuprok programok, amelyek egy vagy több hálózati szolgáltatás szimulált elérését teszik lehetővé géped portjain. Egy esetleges támadó úgy látja, hogy a gép könnyen támadható és valószínűleg fel is törhető a szolgáltatáson keresztül. A mézescsupor arra való, hogy rögzítse a támadó erőfeszítéseit mind portokon történő forgalma, mind a billentyűütései tekintetében. Később ugyanis, ezen információk ismeretében ugyanis jobban fel lehet készülni a támadásokra. Itt le is tölthetsz egy ilyet, neve: Deception Tool Kit. A cél tehát a rögzítés mellett az, hogy elérjük, hogy megtámadjanak, különben nem lesz mit rögzíteni. Így a mézescsuprok akkor eredményesek, ha ismert jól ismert szervereket: webet, mailt vagy DNSt futtatunk, mert ezek azok a rendszerek, amelyeket a leggyakrabban támadnak.

[3] levélszemét

- Az azonosításhoz erős jelszavak alkalmazása és ezek lecserélése hetente/havonta rendszeresen.
- Vezeték nélküli kapcsolat estén robosztus jelszó használata.
- Magas szintű elvárások a fizikai védelemmel szemben.
- Egy optimális hálózati analizátor, vagy monitorozó.

Nagyvállalatok:

- Egy erős tűzfal és proxy a nem kívánatos személyek távoltartására.
- Egy erős antivírus- és internetvédelmi szoftver.
- Az azonosításhoz erős jelszavak alkalmazása és ezek lecserélése hetente/havonta rendszeresen.
- Vezeték nélküli kapcsolat estén robosztus jelszó használata.
- Többszintű fizikai védelem az megelőzés érdekében.
- Egy hálózati analizátor vagy monitorozó készítése és használata, ha szükséges.
- Fizikai védelmi osztály implementációja, mint például egy kamerás megfigyelési rendszer a belépési helyeken és az elzárt zónáknál.
- Biztonsági kerítés a vállalat körül.
- Tűzoltó-készülékek a tűzveszélyes helyekre, mint a szerverszobák és a biztonsági szobák.
- Biztonsági őrök segíthetnek maximalizálni a védelmet.

Kormányhivatalok:

- Egy erős tűzfal és proxy a nem kívánatos személyek távoltartására.
- Egy erős antivírus- és internetvédelmi szoftver.
- Erős titkosítás, általában 256 bites kulccsal.
- Vezeték nélküli kapcsolat esetén lista az engedélyezett felhasználókról, minden más blokkolása.
- Minden hálózati hardver a biztonsági zónában legyen.
- Minden hoszt a privát hálózatban kell legyen, ami kifelé láthatatlan.
- Minden szerver a demilitarizált zónában legyen, vagy egy külső és belső tűzfal között helyezkedjen el.
- Biztonsági kerítés a hivatal körül és a vezeték nélküli kapcsolat korlátozása erre a távolságra.

2. Behatolás-megelőzés

2.1 Incidensek megelőzése

Az incidensek számának alacsonyan tartása nagyon fontos a szervezeteknek, hogy így megvédjék a vállalati folyamatokat. Ha a biztonsági intézkedések nem megfelelőek, akkor a sok incidens elözönli az incidenskezelő csapatot. Ez ahhoz vezethet, hogy lelassul és megszűnik a reakció, ami nagy vállalati károkhoz vezet (például, hosszabb szolgáltatás-lefutási idő, vagy adat-elérhetetlenség). A szervezeteknek fontos, hogy rendszeresen felbecsüljék a kockázati tényezőit a rendszereknek és programoknak. A becslésnek tartalmaznia kell, hogy milyen kockázatok következhetnek a sebezhetőség és fenyegetettség kombinációiból. Minden kockázatot fontossági sorrendbe kell állítani, és a kockázatokat mérsékelhetjük, transzformálhatjuk vagy elfogadhatjuk, míg elérünk egy olyan kockázati szintet, ami elfogadható.

Egy másik előnye a rendszeres kockázatbecslésnek az, hogy kritikus forrásokat azonosítanak, és a munkatársak számára kihangsúlyozott e források figyelése és kezelése. A szervezetnek nem csak ezeket kell figyelnie, hanem a kevésbé veszélyesnek tűnő forrásokat is, mert a szervezet csak annyira biztonságos, amennyire a leggyengébb láncszeme. Jegyezzük meg, attól függetlenül, hogy mennyire hatásos a kockázatelemzés, csak az aktuális kockázatokat világítja meg. Mindig új és új sebezhetőségek és fenyegetettségek keletkeznek, így a számítógépes biztonság egy folyton változó folyamat, ami megköveteli a kellő figyelmet, hogy mindig napra készek legyünk.

- *Patch*[1] *management*: Számos biztonsággal kapcsolatos informatikai kutatás egyetért abban, hogy az incidensek jelentős százaléka a rendszerben és az alkalmazásokban lévő relatív kisszámú sebezhetőségből származik. A nagyvállalatoknak létre kell hozniuk egy, a frissítések telepítésére vonatkozó tervet, hogy segítse a rendszergazdákat a biztonsági javítások azonosításban, tesztelésben és a fejlesztésben.
- *Hosztbiztonság*: Amellett, hogy minden hoszt rendelkezik a szükséges biztonsági javításokkal és frissítésekkel, minden hosztot úgy kell konfigurálni, hogy csak minimális folyamatokat nyújtson, a megfelelő felhasználóknak és hosztoknak. A

[1] Szó szerint sebtapasz, folt. Javítócsomagot jelöl. Ennek segítségével orvosolják a szoftverfejlesztők az eredeti programban annak kiadása után felszínre került programhibákat.

bizonytalan beállításokat meg kell változtatni. Figyelmeztetést kell küldeni a felhasználónak valahányszor biztonságos erőforráshoz próbál hozzáférni. A hosztokon engedélyezni kell a hozzáférést és fel kell jegyezni a jelentős biztonsági eseményeket. Számos szervezet használ operációs rendszer és alkalmazás leírásokat, hogy segítsék az adminisztrátorokat a hosztok eredményes védelmében.

- *Hálózati biztonság*: A hálózat paramétereit úgy kell beállítani, hogy visszautasítsanak minden tevékenységet, ami nincs határozottan engedélyezve. Csak a szükséges tevékenységeket kell engedélyezni, melyek a szervezet megfelelő működéséhez szükségesek. Ez tartalmazza az összes csatlakozási pont védelmét, mint például a modemek, a VPN és a dedikált kapcsolatokat más szervezetekkel.
- *Rosszindulatú kódelhárítás*: Szoftvert kell telepíteni a szervezeten kívül a rosszindulatú kódok detektálására és megállítására, mint például a vírusok, a férgek és a trójai programok. A rosszindulatú kódelhárítást hoszt szinten, az alkalmazás szintű szervereken (e-mail szerver, proxy-k) és az alkalmazás szintű klienseknél kell telepíteni.
- *Felhasználói felkészítés és tréning*: A felhasználóknak el kell magyarázni, hogy hogyan használják helyesen a hálózatot, a rendszert és az alkalmazásokat. A korábban történt incidensek tanulságait is meg kell osztani a felhasználókkal, hogy lássák milyen hatással lehetnek tetteik a szervezetre. Ez a felhasználók tudatosságát növeli az incidensekkel kapcsolatban. Ezáltal az incidensek gyakorisága csökken, különösen azoké, amelyek rosszindulatú kódokból, illetve a biztonsági előírások megszegéséből származnak. Az IT munkatársakat képezni kell, hogy a hálózataik, rendszereik, és alkalmazásaik a szervezetben levő biztonsági szabványokkal összhangban legyenek.

2.2 Incidens kategóriák

Incidensek számtalan úton bekövetkezhetnek, szóval nem praktikus az incidensek lépésről-lépésre való kezelési technikáinak fejlesztése. A legjobb, amit a szervezet tehet, hogy az incidenskezelésre általánosan készül, és a gyakori incidensekre pedig specifikusan.

Főbb kategóriák:

- *Szolgáltatásmegtagadás* – egy támadás, mely a jogosulttól megtagadja a hálózat, a rendszer vagy az alkalmazás hozzáférést az erőforrások leterhelésével
- *Rosszindulatú kód* – egy vírus, féreg, trójai vírus vagy más rosszindulatú kód, ami megfertőzi a hosztot

- *Jogosulatlan hozzáférés* – egy személy logikai vagy fizikai hozzáférést szerez engedély nélkül a hálózathoz, a rendszerhez, alkalmazáshoz, adathoz vagy más számítástechnikai erőforráshoz
- *Helytelen felhasználás* – egy személy megszegi az elfogadott hálózati-, vagy számítógép használati előírásokat
- *Többszörös komponens* – egy olyan incidens, ami több incidenst foglal magába

2.3 Az incidens jelei

Nagyon sok szervezet számára, az incidens kezelési folyamat legnagyobb kihívást jelentő része az incidensek pontos detektálása, a lehetséges incidensek felbecslése, és ha történt valamilyen incidens, akkor az milyen típusú, fokú és jelentőségű. Hogy miért jelent ez ilyen kihívást, az három tényező kombinációjából áll össze:

- Az incidenseket sokféle képen lehet detektálni. Az automatikus detektáló képesség, hálózat alapú és hoszt alapú behatolás érzékelő és megelőző rendszerek, antivírus szoftver, és log[1] analizáló együttese. Az incidensek manuálisan is érzékelhetők, mint például a felhasználók által jelentett incidensek. Néhány incidensnek nyilvánvaló jelei vannak, így könnyen detektálhatóak, míg mások szinte észrevehetetlenek automatika nélkül.
- A potenciális incidensek száma tipikusan magas; például egy szervezetnél nem szokatlan, ha riasztások ezreit, vagy milliót kapják naponta.
- Mély, specifikus tudás és tapasztalat szükséges a megfelelő és eredményes incidenshez kapcsolódó információ elemzéséhez. A legtöbb szervezetben, az a pár ember, akik ilyen szintű tudással rendelkeznek, valószínűleg más munkára vannak kijelölve.

Az incidensek jelei a két kategória valamelyikébe esnek: megelőző jelek és előjelek. A megelőző jel egy olyan jel, amit az incidens a jövőben okozni fog. Az előjel pedig egy olyan jel, amit az incidens okozott, vagy épp okoz. Túl sok típusa van az előjeleknek, hogy kimerítő listát készítsenek róla, de néhány példa ezekre:

- A hálózati behatolás-érzékelő szenzor jelez, ha puffer túlcsordulás veszélye áll fenn egy FTP szerveren.

[1] naplózás

- Az antivírus szoftver jelez, ha egy hoszt féreggel fertőzött.
- A webszerver összeomlott.
- A felhasználó jelez, ha túl sokáig tart az internethez való hozzáférés.
- A rendszergazda olyan fájlnevet lát, ami furcsa karaktereket tartalmaz.
- A felhasználó jelez az internetes technikai ügyfélszolgálatnak fenyegető e-mail üzenet esetén.
- Egy alkalmazás sok helytelen bejelentkezési kísérletet tapasztal egy ismeretlen távoli rendszerről.
- A rendszergazda sok visszaküldött üzenetet lát, gyanús tartalmakkal.

Ne higgyük azt, hogy az incidensdetektálás szigorúan csak kémkedésből áll. Néhány esetben, a szervezet érzékelni tudja az incidensgyanús aktivitásokat. Például, egy hálózati IDPS[1] szenzor érzékelhet szokatlan port-szkennelést, ami a hosztok egy csoportját célozza, közvetlenül egy DoS[2] támadás indítása előtt történhet a csoportban lévő hosztok valamelyike ellen. A szkennelési aktivitáshoz kapcsolódó behatolás-érzékelő riasztások megelőző jelzésként szolgálhatnak egy későbbi DoS támadáshoz. További példák a megelőző jelzésekre:

- A webszerver naplóbejegyzéseit (log) figyeli a web sebezhetőségét vizsgáló eszköz
- Egy közlemény egy új biztonsági résekről és sebezhetőségekről, ami a szervezetek levelezési szervereinek sebezhetőségét használja ki.
- Egy hacker csoport fenyegetése azt jelenti, hogy megtámadhatják a szervezetet.

Nem minden támadást lehet észlelni megelőző jelzésekkel. Néhány támadásnak nincs semmi előjele, amíg mások olyan előjeleket generálnak, amit a szervezet nem ismer fel. Amikor egy megelőző jelet detektálnak, a szervezetnek lehetősége van arra, hogy megelőzze az incidenst, a biztonsági állapot automatikus, vagy manuális megváltoztatásával, hogy megmentse a célt a

[1] Intrusion Detection and Prevention System - behatolás-érzékelő és -megelőző rendszer

[2] Denial of Service – szolgáltatásmegtagadásos támadás. A szolgáltatás ismételt igénybevétele, melynek révén a szolgáltatás túlterhelésre kerül vagy leáll. Ennek technikája a számítógép erőforrásait újra és újra igénylő program végrehajtása (erőforrás lehet a memória, CPU, TCP-UDP kapcsolat, diszkterület)

támadástól. Bizonyos esetekben úgy kell cselekednie a szervezetnek, mintha már bekövetkezett volna az incidens, így a kockázat gyorsabban csökkenthető.

3. Behatolás-érzékelés

3.1 Behatolás-érzékelő rendszerek (IDS)

Egy behatolás-érzékelő rendszer alapvetően számítógépes rendszerek nem kívánatos manipulációit érzékeli, főleg az interneten keresztülieket. A manipuláció általában cracker-ek támadásait takarja. Egy IDS nem tudja érzékelni a titkosított adatfolyamokban a támadásokat.

Egy behatolás-érzékelő rendszer különböző, szándékos rongáló magatartások felismerésére használható, melyek a számítógépes rendszer megbízhatóságát és védelmét ingatják meg. Ez a gyenge szolgáltatások elleni támadásokat tartalmazza. Például a programok adatbevitelét kihasználó támadás, a hosztok elleni támadások, mint jogosultságok kiterjesztése, jogosulatlan bejelentkezések, védett fájlokhoz való hozzáférés és kártékony programok (malware, vírusok, férgek és trójai vírusok).
Egy behatolás-érzékelő rendszer különböző részekből áll: *szenzorokból*, melyek a védelmi eseményeket generálja, *konzolból*, ami megjeleníti a biztonsági eseményeket és irányítja, riasztja a szenzorokat, és áll még egy *központi egységből*, ami feljegyzi egy adatbázisba a szenzorok által érzékelt eseményeket. A központi egység egy szabályrendszert használ, ami alapján riasztásokat ad le biztonsági események érzékelésekor. Egy IDS besorolására sok lehetőség van a szenzorok helyétől és a központi egység riasztásának módjától függően. A legtöbb egyszerű IDS konfigurációkban mindhárom egység egyetlen eszközbe van telepítve.

3.2 A behatolás-érzékelő rendszerek fajtái

Egy hálózat alapú behatolás-érzékelő rendszerben az érzékelők a hálózat azon csomópontjainál vannak elhelyezve, melyeket megfigyelni szeretnénk. Ez gyakran a demilitarizált zónában, vagy a hálózat határain van. Az érzékelők az egész hálózati forgalmat figyelik és minden egyes csomagot megvizsgálnak rosszindulatú tartalmak után kutatva. A rendszerben az adatforgalmat és a protokollok megfelelőségét PIDS[1] és APIDS[2] segítségével figyelik. A hoszt alapú rendszerekben az érzékelő gyakran tartalmaz egy szoftvert, ami a felhasználó minden tevékenységét feljegyzi, ahol ez telepítve van. A két rendszernek létezik hibrid változata.

---

[1] Protcol-based Intrusion Detection System - protokoll alapú behatolás-érzékelő rendszer

[2] Application Protcol-based Intrusion Detection System - alkalmazás protokoll alapú behatolás-érzékelő rendszer

- A *hálózat alapú behatolás-érzékelő rendszer* (NIDS[1]) platform független. Több kliens forgalmát figyeli a hálózaton és ez alapján azonosítja a behatolásokat. A hálózat alapú behatolás érzékelő rendszer porttükrözéssel fér hozzá a hálózati forgalomhoz, mégpedig úgy, hogy egy hubhoz, vagy switchhez csatlakoztatjuk és lemásoljuk a hálózati forgalmat. Egy ilyen rendszerre példa a Snort.
- *A protokoll alapú behatolás-érzékelő rendszer* (PIDS) egy rendszerből, vagy egy ügynökből áll, ami a szerver oldalon figyeli és vizsgálja a kommunikációs prokollt a két csatlakozott eszköz (felhasználó/ PC, vagy rendszer) között. Ez egy webszerver esetében a HTTPS protokoll adatfolyamának monitorozását jelenti, ami a védeni kívánt rendszerhez tartozó HTTP protokollt figyeli. Ahol HTTPS-t használnak, ott ezt a rendszert oda kell elhelyezni, ahol még a HTTPS titkosítatlan és nem került bele a web megjelenítési rétegbe.
- Az *alkalmazás protokoll alapú behatolás-érzékelő rendszer* (AIPDS) egy rendszerből, vagy egy ügynökből áll, ami leginkább több szerver között helyezkedik el. Feladata, hogy megfigyelje és vizsgálja az alkalmazás specifikus protokollok közötti kommunikációt. Például egy adatbázissal rendelkező webszerver esetében az SQL protokollt fogja figyelni, speciálisan középszintű és cégszintű bejelentkezéskor, a bejelentkezési tranzakció és az adatbázis közötti műveletet.
- A *hoszt alapú behatolás-érzékelő rendszer* (HIDS[2]) egy ügynökből áll egy hoszton. A rendszer képes azonosítani a behatolásokat rendszerhívások, programok naplófájljainak, a fájlrendszer módosítása és más egyéb felhasználói tevékenységek vizsgálatával. Egy példa erre a rendszerre az OSSEC.
- A *hibrid behatolás-érzékelő rendszer* (Hybrid IDS) kettő, vagy több különböző megközelítést ötvöz. A hosztügynök információi hálózati információkkal vannak ötvözve egy átfogó hálózati nézetté. Egy példa a Hybrid IDS-re a Prelude.

3.3 Passzív rendszer kontra kémlelő rendszer

Egy passzív rendszerben az IDS szenzorai a potenciális réseket detektálják, feljegyzik az információkat és jelet bocsátanak ki a konzolra és/vagy a tulajdonosnak. Egy kémlelő rendszerben, ami más néven behatolás-megelőző rendszerként (IPS- Intrusion Prevention

[1] Network Intrusion Detection System

[2] Host-based Intrusion Detection System

System) is ismert, az IPS válaszol a gyanús folyamatnak a kapcsolat bontásával, vagy a tűzfal átállításával, hogy blokkolja a forgalmat a gyanúsnak tartott forrástól. Ez történhet automatikusan, vagy az operátor közreműködésével.

Ámbár mindkettő a hálózati biztonsághoz tartozik, egy IDS abban különbözik egy tűzfaltól, hogy egy tűzfal kívülről keres behatolásokat, hogy megakadályozza azokat, míg a tűzfalak korlátozzák a hálózatok közötti hozzáférést a behatolás megelőzése miatt, és nem jelzik a hálózaton belüli támadásokat. Egy IDS felépíti magában a kapcsolatot és csak azokat a csomagokat engedi be a hálózatra, melyeket biztonságosnak tart. Az IDS által meghozott döntéseket négy csoportba lehet osztani:

- Hamis negatív (fals negativ): ebbe a csoportba azok a csomagok kerülnek, amiket az IDS rosszindulatúnak értékelt, de nem az.
- Hamis pozitív (fals positive): ebbe a csoportba azok a csomagok kerülnek, amiket az IDS helyesen nem engedett be a hálózatba.
- Igaz negatív (true negativ): ebbe a csoportba azok a csomagok kerülnek, amiket az IDS biztonságosnak talált, de valójában nem az.
- Igaz pozitív (true positive): ebbe a csoportba azok a csomagok kerülnek, amiket az IDS helyesnek értékelt és az is.

Az egy csoportba eső csomagok száma attól függ, hogy az IDS mennyire van érzékenyre állítva. Az IDS a hálózaton belülről származó támadásokat is figyeli.

IDS kikerülési technikák

A behatolás-érzékelő rendszer kikerülési technikák igyekeznek elkerülni az érzékelést különböző állapotok létrehozásával az IDS-en és célzott számítógépen. A támadó ezt a gyakorlatban, vagy magának a támadásnak, vagy a hálózati forgalomnak, mely a támadást tartalmazza, manipulálásával éri el.

4. Eseménykezelés

Az eseménykezelésnek számos fázisa van, a kezdeti felkészüléstől az esemény bekövetkezése utáni elemzésig. A kezdeti fázis tartalmazza az eseménykezelő csapat kiválasztását, valamint kiképzését, és a szükséges eszközök, valamint források beszerzését. Az előkészületek alatt a szervezet megpróbálja korlátozni a bekövetkező incidensek számát úgy, hogy kiválasztanak és implementálnak olyan ellenőrzéseket, melyek kockázatelemzésen alapszanak. Akárhogy is, az ellenőrzések implementálása után is marad némi kockázat, mivel egy ellenőrzés sem tökéletes. A biztonsági rések detektálása legalább annyira fontos a szervezetnek, mint egy bekövetkezett incidens kezelése. Az incidens komolyságától függően, a szervezet enyhíteni tudja, vagy teljesen helyre tudja hozni a károkat. Miután az incidenst megfelelően kezelték, a szervezet készít egy tanulmányt, ami tartalmazza az esemény részleteit, az okozott károkat és a lépéseket, melyeket a szervezetnek tennie kell, hogy megakadályozza a jövőben a bekövetkezését. A fő fázisai az incidenskezelési folyamatnak az előkészülés, detektálás és analizálás, behatárolás/megsemmisítés/visszaállítás, és az esemény bekövetkezése utáni elemzés.

4.1 Előkészülés

Az incidenskezelési módszerek nem csak magát az incidenskezelés képességét hangsúlyozzák ki, hanem az incidensek megelőzésének képességét is, hogy a szervezet reagálni tudjon az incidensre. Ezáltal a szervezet rendszere, hálózata és a programjai biztonságossá váljanak. Bár az incidenskezelő csapat nem igazán felelős az incidensek megelőzéséért, de mára már annyira fontos, hogy alapvető részét képezi az incidenskezelő tervnek. A biztonsági rendszer elkészítésénél az incidenskezelő csapatok szaktudása nagy jelentőséggel bír.

Felkészülés az incidenskezelésre:

Eszközök és erőforrások:

- Kapcsolati információk (telefonszámok, e-mail címek, nyilvános kulcsok)
- Telefonos információk
- Incidensértesítési mechanizmusok (telefonszámok, e-mail címek, online fórumok, és olyan mechanizmusok, ahol névtelenül lehet jelenteni az incidenst)
- Személyhívók, mobiltelefonok
- Titkosító szoftver

Incidensanalizáló szoftver és hardver:

- Helyreállító eszközök
- Laptopok
- Pótmunkaállomások, szerverek és hálózati berendezések
- Háttértárak (floppy, CD, DVD)
- Hordozható nyomtató
- Csomag és protokollvizsgálók
- Mozgatható háttértárak
- Bizonyítékgyűjtő alkatrészek

Incidensanalízis erőforrások:

- Port lista
- Dokumentációk
- Hálózati diagrammok és a kritikus eszközök listája (Web, e-mail, adatbázisszerverek)
- Alapkonfiguráció
- Kriptográfiai hash-ek
- Biztonsági javítások és frissítések
- Helyreállítási képek (operációs rendszernek, programoknak)

Számos incidenskezelő csapat készít „jump kit[1]”-et (szó szerint urgró felszerelés), ami egy hordozható csomag. A „jump kit” olyan anyagokat tartalmaz, amire egy incidenskezelőnek szüksége lehet vizsgálatkor. A „jump kit” mindig használható, tehát amikor egy súlyos incidens bekövetkezik, az incidenskezelőnek rendelkezésére áll egy eszköz az incidens kezelésére. A „jump kit” tartalmazhat például, visszaállító eszközöket, háttértárat, központi hálózati eszközt, operációs rendszert, programokat és patch-eket (szó szerint tapasz). Mivel egy „jump kit” használata jelentősen felgyorsítja a reakciókat, ezért a csapatnak célszerű a „jump kit” használata. Szintén nagyon fontos a „jump kit” karbantartása (például biztonsági frissítések telepítése laptopokon, operációs rendszer frissítése). A szervezeteknek támogatniuk kellene a „jump kit”-ek összeállítását, mert így sokkal gyorsabban és hatékonyabban lehet kezelni az incidenseket.

[1] Szó szerint urgró felszerelés. Egy hordozható csomag, vagy eset, ami olyan anyagokat tartalmaz, amire egy incidenskezelőnek szüksége lehet vizsgálatkor

5. Hálózatok

5.1 A hálózat felderítése

A hálózat feltérképezése nagyon fontos feladata a hálózat kezelőjének. Tudnia kell, hogy milyen eszközök, és hogyan csatlakoznak a hálózathoz, hogy pontos képet kapjon a hálózatról. Ha megfelelő képpel rendelkezik, sokkal könnyebben tudja karban tartani, javítani, ha valamilyen hiba történne, valamint egyszerűbben végrehajthatóak a frissítések, így csökkenthetőek a behatolás veszélyei.
Lényeges, hogy miközben felderítjük a hálózatot, ne terheljük le nagyon azt. Az algoritmus, amit használunk, legyen gyors, dolgozzon kevés hibával és detektálja a hálózathoz csatlakozó eszközöket.
Több mód van a hálózat feltérképezésére:

- SNMP scan
- IP tartomány scan
- ping segítségével
- traceroute
- DNS felhasználásával (gyors de nem mindig elérhető)
- broadcast ping

Egy alapvető módszer lehet, ha a lehetséges címek egy részével kezdjük és ping segítségével megvizsgáljuk, hogy létezik-e a hoszt. Ezután traceroute segítségével megadhatjuk a kapcsolatot a hosztok között, így feltérképezhetjük a hálózatot.

5.2 Hálózati forgalom analizálása

A hálózati IDS eszközök passzív hálózati megfigyelést végeznek, hogy detektálják a lehetséges fenyegetéseket. Passzív monitorozással a rendszergazda, teljesen megértheti a hálózati topológiát, hogy milyen szolgáltatások érhetőek el, milyen operációs rendszereket használnak, és hogy milyen sebezhetőségek fordulhatnak elő. A legtöbb ilyen információ automatizált módon megszerezhető. Ehhez egy csomaganalizáló eszközre van szükségünk, amilyen a tcpdump, snoop, vagy a Wireshark. A csomagok útjainak analizálásával, valamint az IP fejrészben található TTL mező segítségével meghatározhatjuk a hálózat logikai felépítését.
Az operációs rendszerek normális esetben véges számlálót állítanak be, 32, 64, 128, 255. Ezt a számot minden router csökkenti, amikor áthalad rajta a csomag.

Passzív operációs rendszerazonosítás

A passzív operációs rendszer (OS[1]) azonosítás alapja az, hogy minden OS IP implementációja eltérő. Ez leginkább a TCP SYN csomagok elemzésével végezhető, mert ezekben a csomagokban van a legnagyobb eltérés. A mezők, amiket ilyenkor figyelnek: a TTL kezdőértéke, a Don't fragment mező, a SYN csomag kezdő mérete, IP ID, TOS mező, forrásport és TCP opciók, mint ablakméret, MSS méret, Window Scaling opció, SackOK opció.

Vannak automatikus eszközök, amik analizálják a csomagokat. Ilyen a Siphon (amit egy ideje nem fejlesztenek), és a p0f (amit folyamatosan fejlesztenek). A p0f inkább az IP mezők és a SYN csomagok kombinációjából következtet, míg a Siphon a TTL, és Don't fragment bit-ből következtet.

Viszonylag egyszerű az elfogott csomagokon ilyen analízist automatizálni. Figyelnünk kell arra is, hogy bizonyos operációs rendszereken meg lehet változtatni a rendszer viselkedését. Linuxon megváltoztathatjuk az IP viselkedését, a sysctl, vagy a /proc/sys/net/ipv4 könyvtárban az értékek megváltoztatásával. Bizonyos opciókat lehet így megváltoztatni, mint például az alapértelmezett TTL értéket, az ablakméret, és más opcionális operációs rendszer specifikus információkat.

Raw socketek[2]

Ha forgalmat vizsgálunk szükséges egy alaptudás, hogy a legtöbb IP stack hogyan működik, és hogy mit veszünk normális forgalomnak, és mit nem. A raw socket-ek képességeinek megértése, és ezeknek a használata, valamint a hálózati forgalomra vonatkozó kihatásuk kulcs fontosságú lehet. A normális socket API kezeli a legtöbb adatot, interfész kiválasztást, TTL értékek beállítása, Sequence és Ack számok és így tovább. Az egyetlen út, hogy könnyen manipuláljuk ezeket a sysctl, vagy egy olyan eszköz használata, ami a csomagokat a raw socket API segítségével rakja össze.

A raw socket API segítségével manipulálhatjuk a csomagok fejrészét, habár ehhez egy kis hálózati proramozási ismeret szükséges. Ha raw socket segítségével akarunk csomagokat készíteni, akkor az összes fejrészben levő mezőt be kell állítani.

---

[1] Operating System

[2] A raw socket egy olyan socket, ami közvetlen hozzáférést biztosít a csomagok fejrészéhez.

A socket pedig kommunikációs kapcsolódási végpont

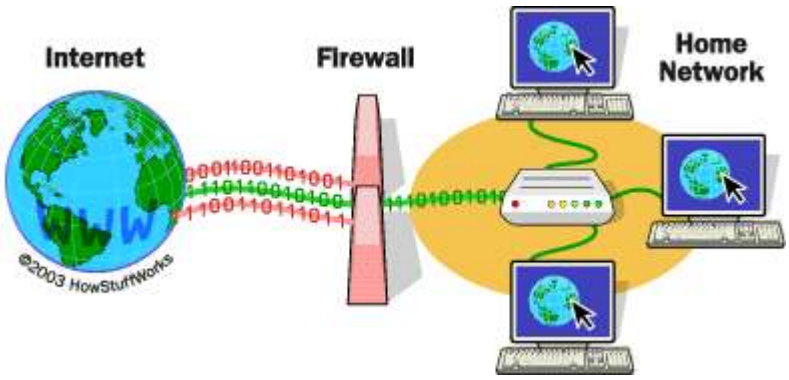
Egy hálózati adatforgalom hozzásegíthet, hogy megértsük a hálózat működését, hogy milyen eszközök, szolgáltatások, és sebezhetőségek vannak. Ezeknek az adatoknak az analízise bizonyos esetekben automatizálható egyszerű script-ekkel. A kapott adatokat felhasználhatjuk, hogy megtudjuk, hogy mennyire vannak jól beállítva a hozzáférési házirendek, és észrevehetjük a hálózatban a sebezhetőségeket.

6. Tűzfalak

A tűzfal megvédi a belső hálózat gépeit, a külső behatolásoktól vagy DoS támadásoktól. Lehet hardver, vagy szoftver, ami egy biztonságos gépen fut. Legalább 2 interfésszel rendelkezik, az egyik a külső a másik pedig a belső hálózat felé, amit meg akar védeni. A tűzfal egy csomópontban, vagy a két hálózat közötti átjáróként található meg, általában egy magánhálózat, és az internet között. A korábbi tűzfalak csak routerek voltak. A tűzfal kifejezés onnan származik, hogy régen a hálózatokat több fizikailag szétválasztott részhálózatokra bontották, hogy ha valamilyen probléma adódna, akkor az ne terjedjen át az egész hálózatra. Az ilyen esetekben a tűzfal megakadályozza a továbbterjedést, mint a valóságban a tűzfalak, tűzálló ajtók a tűz továbbterjedését.

Egy tűzfal a két hálózat közötti forgalmat figyeli, és bizonyos kritériumok alapján vizsgálja. Ha megfelel a forgalom, akkor átirányítja a két hálózat között, ha nem, akkor megállítja. A kimenő és bejövő forgalmat is szűri. Külső hozzáférést is tud engedélyezni belső hálózati erőforrásokhoz, például hoszt alkalmazásokhoz. Arra is használhatjuk, hogy naplózzon minden belépést a magánhálózatba, és figyelmeztessen, ha illetéktelen hozzáférés történik.

A tűzfalak a csomagokat forrás-, célcím és portszámok alapján szűrhetik. Ezt úgy ismerjük, hogy címszűrés. Képesek különböző hálózati forgalom szűrésére is, ezt protokollszűrésnek is nevezik, mert a döntés, hogy továbbítja, vagy visszautasítja a csomagokat, attól függ, hogy milyen protokollt használ. Például HTTP, FTP.



Mire nem használható egy tűzfal?

Az alkalmazottak gondatlanságát, vagy rosszhiszemű magatartását nem lehet tűzfalakkal irányítani. A jelszavak helyes, és helytelen használatának házirendjét be kell tartani. Ezeket a dolgokat a biztonsági házirend tervezésénél figyelembe kell venni, nem lehet csak tűzfalakkal megoldani. A tűzfalak nem védenek meg az úgynevezett szociális manipuláció (Social Engineering) támadások ellen, amik az emberi hiszékenységet, és bizalmat használják ki.

Kinek van szüksége tűzfalra?

Mindenkinek szüksége van egy tűzfalra, aki felelős egy belső hálózatért, ami egy publikus hálózathoz tartozik.

Hogyan működnek?

Két alapvető módszert használ a tűzfal. Engedélyezi a forgalmat, ha megfelelt a kritériumoknak, vagy tiltja, amennyiben nem. A tűzfalak a forgalom tartalmát, vagy a forrás-, célcímeit és a portokat figyelik. Bonyolult szabályokat is használhat az alkalmazás adatforgalmának vizsgálatához, hogy eldöntse átengedi-e, vagy sem. Az, hogy egy tűzfal milyen típusú forgalmat enged át, attól függ, hogy a hálózat melyik rétegében működik.
Egy professzionális tűzfal elkap minden csomagot mielőtt az operációs rendszerhez érkezne.

Típusai:

A tűzfalak 4 nagyobb kategóriába sorolhatók:

- csomagszűrő
- körkörös szintű átjárók
- alkalmazási rétegbeli átjárók
- alapos, több szinten vizsgáló tűzfalak.

A csomagszűrő tűzfalak az OSI modell hálózati rétegében működnek, vagy a TCP/IP modell IP rétegében. Általában egy router része. Attól függően, hogy a csomag megfelel-e a kritériumoknak a tűzfal átengedheti, átirányíthatja, üzenetet küldhet a feladónak, vagy eldobhatja a csomagokat. A szűrési kritériumok tartalmazhatják, a célcímet, forráscímet, portszámokat, vagy a használt protokollt. Nagy előnyük az ilyen routereknek, az olcsóságuk, és hogy kevésbé avatkoznak bele a hálózati forgalomba.

Körkörös szintű átjárók

Ezek a tűzfalak az OSI modell viszony rétegében működnek. TCP kézfogásokat figyelik, hogy kiszűrjék az illegális forgalmat. Ez hasznos, ha el akarjuk rejteni az információkat a védett hálózatról. Más részről, nem szűrik a csomagokat külön-külön.

Alkalmazási rétegbeli átjárók

Ezeket az átjárókat gyakran hívják proxynak. Hasonlítanak a körkörös átjárókra, a csomagokat alkalmazásszinten tudják szűrni. A kimenő és bejövő csomagok nem érhetnek el olyan szolgáltatást, amihez nincsen proxy. Például, ha az alkalmazást úgy konfiguráljuk, hogy web proxy legyen, akkor nem fog engedélyezni semmilyen FTP forgalmat. Ezeket nem lehet kiszűrni csomagszűrést végező tűzfalakkal. Az alkalmazásszintű átjárók képesek arra is, hogy a felhasználók bejelentkezését és tevékenységét naplózzák. Magas szintű biztonságot eredményeznek, de jelentősen kihatnak a hálózat teljesítményére. Lassítják a hálózatot a

folyamatos tartalomszűréssel. A felhasználók felé nem láthatatlanok és minden kliens gépen be kell állítani.

Alapos, több szinten vizsgáló tűzfalak

Ezek a tűzfalak az előző három típus kombinációi. A csomagokat szűrik először, majd a viszony rétegben és az alkalmazási rétegben is megvizsgálják a csomagokat. Az alkalmazási rétegbeli adatok felismerése algoritmusokon alapszik, így a tűzfalak magas biztonságot nyújtanak jó teljesítmény mellett, és a végfelhasználók felé láthatatlanok. Az ilyen típusú tűzfalak ára elég magas és a bonyolultságukból fakadóan megfelelő képzettségű szakember szükséges a felkonfigurálásukhoz, hogy elérjük a kívánt biztonsági szintet. A tűzfalak megvédik a magánhálózatokat a behatolásoktól az internet felöl, de egy tűzfal önmagában kevés a biztonság kiépítéséhez. Lehetőséget biztosítanak a hálózati rendszergazdák számára, hogy az internet bizonyos szolgáltatásaihoz biztosítsák a hozzáférést a magánhálózatban levő felhasználók számára. Az, hogy milyen szolgáltatásokat engedélyeznek, alapvető része bármely információkezelési programnak, amibe bele tartozik, hogy nem csak a személyes információt védjük meg, hanem azt is, hogy kinek mihez van hozzáférési joga. A jogosultságokat inkább a munkaköri leírás alapján kell kiosztani, mint a mindent vagy semmit alapon.

7. DMZ[1]

A számítógépes biztonságban, a demilitarizált zóna (DMZ- demilitarized zone) kifejezés, a katonai használatból ered, de inkább határzónának, vagy kerület hálózatnak felel meg, Egy fizikai, vagy logikai alhálózat, ami egy szervezet külső szolgáltatásait tartalmazza egy nagyobb, megbízhatatlan hálózat felé, általában az internet felé. A DMZ célja, egy tovább biztonsági réteg létrehozása a szervezetek helyi hálózatához, így egy külső támadó csak a demilitarizált zónában lévő eszközökhöz fér hozzá, ami jobb, mintha az egész hálózathoz férne hozzá.

Alapok

Egy hálózatban azok a hosztok a legsebezhetőbbek, amelyek valamilyen szolgáltatást nyújtanak a helyi hálózaton kívüli felhasználóknak, mint például levelezés, web és DNS szerverek. Annak következtében, ahogy ezeknek a hosztoknak a veszélyeztetettsége nő, egy saját alhálózatba kerülnek, hogy megvédjék a hálózat többi részét az esetleges sikeres támadásoktól. Fontos, hogy a demilitarizált zónában levő hosztok ne legyenek képesek közvetlen kapcsolatot létesíteni a belső hálózatban lévő bármely hoszttal, de a DMZ- n belüli és a külső hálózatba irányuló kommunikáció engedélyezett legyen. Ez teszi lehetővé a demilitarizált zónában levő hosztoknak, hogy szolgáltatást nyújtsanak, mind a hálózaton kívülre, mind a hálózaton belülre, mivel egy közbülső tűzfal irányítja a forgalmat a DMZ szerverek és belső hálózatban levő hosztok között.

7.1 A demilitarizált zónába tartozó szolgáltatások

Alapvetően minden olyan szolgáltatás, ami a külső hálózatban levő felhasználókhoz tartozik, a demilitarizált zónában kell, hogy legyen. A leggyakoribb ilyen szolgáltatások a webszerverek, levelezési szerverek, FTP szerverek és DNS szerverek. Néhány szituációban további lépéseket kell tenni, hogy megvédjük a szolgáltatásokat.

Webszerverek:

A webszervereknek egy belső adatbázissal kell kommunikálniuk, hogy egy-két speciális szolgáltatást nyújtsanak. Mióta az adatbázis szerver nem érhető el publikusan és érzékeny információkat tartalmazhat, azóta nem tanácsos ezt a demilitarizált zónában tárolni. Általában nem jó ötlet, hogy a webszervert közvetlenül hagyjuk kommunikálni a belső adatbázissal.

[1] demilitarized zone - demilitarizált zóna

Ehelyett egy alkalmazásszervert használhatunk kommunikációs médiumként a webszerver és az adatbázis között. Ez valószínűleg sokkal bonyolultabb, de egy újabb biztonsági réteget hoz létre.

E-mail szerverek:

Pontosan az e-mail-ek személyes jellege miatt nem jó ötlet ezeket a demilitarizált zónában tárolni. Ehelyett az e-mail-eket egy belső e-mail szerveren kell tárolni. A demilitarizált zónában levő levelező szerver a bejövő leveleket a belső levelező szervnek kell hogy továbbítsa, és a belső levelező szerver a kimenő leveleket a külső levelező szervernek kell hogy továbbítsa. Ideális esetben minden kommunikációt a belső levelezési szervernek kell kezdeményeznie.

Proxy szerverek:

Biztonságtechnikai és monitorozási okok miatt vállalati környezetben, ajánlott egy proxy szerver telepítése a demilitarizált zónába. Ennek a következő előnyei vannak:

- Rákényszeríti a belső felhasználókat a proxy pontos használatára, hogy hozzáférjenek az internethez. Nem szabad megengedni a felhasználóknak, hogy közvetlenül csatlakozzanak az internethez és kikerüljék a demilitarizált zóna védelmét.
- Lehetővé teszi a társaság számára, hogy sávszélességet spóroljon meg, mivel néhány webes rész fennakad a proxy szerveren.
- Lehetővé teszi a rendszergazdának, hogy rögzítse és nyomonkövesse a felhasználók tevékenységeit és biztos lehessen benne, hogy semmilyen illegális tartalmat nem töltenek le, vagy fel a munkavállalók. Például számos EU országban a társaság igazgatója felelős a munkatársak internetes tevékenységeiért.

Fordított proxy szerverek:

Egy fordított proxy szerver ugyanarról gondoskodik, mint egy proxy szerver, de a másik irányba. A belső felhasználóknak nyújtott szolgáltatások helyett, közvetett hozzáférést nyújt a belső erőforrásokhoz a külső hálózatból. (Általában az internethez.) Egy backdoor szolgáltatás, mint amilyen a levelezési rendszer, nyújtható a külső felhasználónak, hogy elolvassa a leveleit, míg a vállalaton kívül tartózkodik, de a távoli felhasználónak nincs közvetlen hozzáférése az e-mail szerveréhez. Egyedül a fordított proxy szervernek van fizikai hozzáférése a belső levelezési szerverhez. Ez egy extra biztonsági szint, ami kiváltképpen fontos, ha belső erőforrásokhoz akarunk hozzáférni kívülről. Általában az ilyen fordított

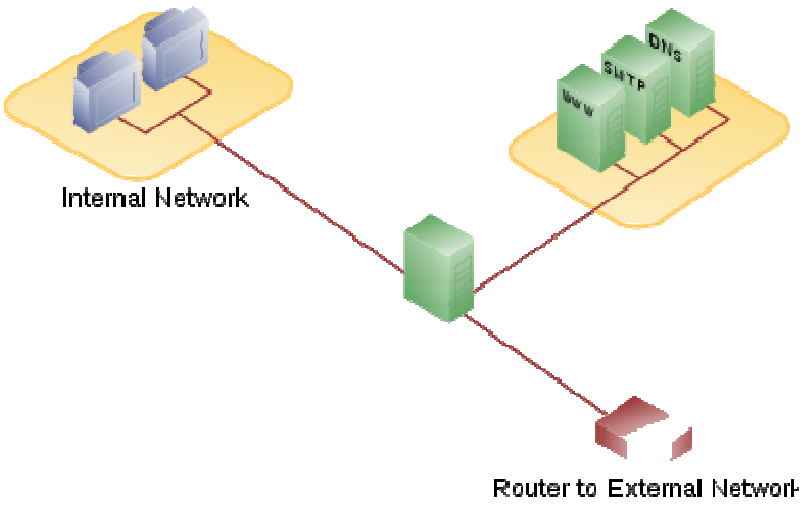
proxy mechanizmus egy alkalmazás szintű tűzfal által biztosított, amikor csak a speciális részére koncentrálnak a forgalomnak.

7.2 Architektúra

Sok módja van, hogy létrehozzunk egy demilitarizált zónával rendelkező hálózatot. A legalapvetőbb módszerek, vagy egyedülálló tűzfalból (háromlábú modellként is ismert), vagy dupla tűzfalból állnak. Ezek az architektúrák kiterjeszthetőek nagyon komplex architektúrákká attól függően, hogy mik a hálózati követelmények.

Egyedülálló tűzfal:

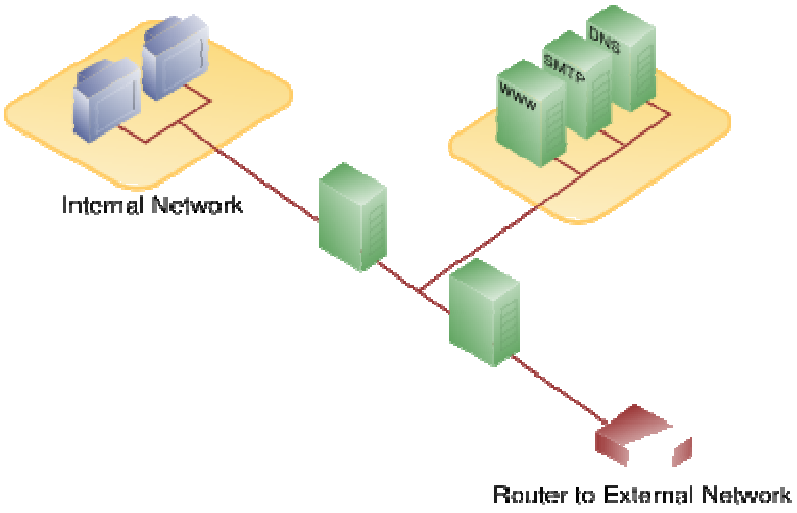
Egy egyedülálló tűzfal, ami minimum 3 hálózati interfésszel rendelkezik, felhasználható egy demilitarizált zónát tartalmazó hálózati architektúra létrehozására. A külső hálózat a szolgáltatótól jön a tűzfalhoz az első interfészen, a belső hálózat a második interfészhez csatlakozik, és a demilitarizált zóna a harmadik interfészhez csatlakozik. A tűzfal lesz az egyetlen hibapontja a hálózatnak és kezelnie kell minden a demilitarizált zónán és a belső hálózaton átmenő forgalmat.



Dupla tűzfal: (szendvicsmodell néven ismert)

Jobb biztonsági megközelítés két tűzfalat használni egy demilitarizált zóna létrehozásához. Az első tűzfalat úgy kell konfigurálni, hogy a demilitarizált zónának címzett mindkét forgalmat, valamint a belső hálózat forgalmát is engedélyezze. A második tűzfalat úgy kell konfigurálni, hogy csak a belső hálózatnak címzett forgalmat engedélyezi a demilitarizált zónából. Az első tűzfalnak sokkal nagyobb hálózati forgalom kezelésére kell képesnek lennie, mint a második tűzfalnak. Elengedhetetlen, hogy a két tűzfal különböző típusú legyen. Ha egy támadónak

sikerül áttörnie az első tűzfalon, sokkal több időbe fog telni a másodikon való átjutás. Ez az architektúra jóval költségesebb, de a megnövekedett biztonság mindenképp kifizetődő.



8. Eszközök

8.1 Nessus:

A Nessus vezető az aktív, sebezhetőségeket detektáló programok között. Kliensből és szkennerből áll. A kliens egy grafikus felületet biztosít. A kliens a hálózat bármely rendszeréről elérheti a szkennert.

Felépítése megengedi, hogy a felhasználó a saját hálózatára szabja a detektálásokat. Gyakran frissülő adatbázissal rendelkezik, ami segít, hogy a legújabb sebezhetőségeket is megtalálja. Különböző kiegészítésekkel bővíthetjük a funkcióit, amiket mi magunk is készíthetünk.

Főbb tulajdonságai:

- gyors felderítés
- érzékeny adatfelderítés, és sebezhetőség analizálás
- használható egy szervezet teljes hálózatában, DMZ-ben, vagy fizikailag elhatárolt hálózatokban is
- ad-hoc[1] detektálást, napi felderítéseket, és gyors auditokat[2] is támogat
- ha a biztonsági központtal irányítjuk, akkor javaslatokat küldhet sebezhetőségekről a megfelelő személyeknek és a biztonsági frissítések és javítások is ellenőrizhetőek
- a detektálás eredményei és paraméterei eltárolhatók, így később is ki lehet elemezni
- a grafikus felület azonnal jelzi az eredményeket, így nem kell megvárni a detektálás befejezését
- támogatja a szolgáltatásdetektálást akkor is, ha nem az alapértelmezett porton fut
- a Nessus szkenner és kliens között kódolt az adatforgalom
- biztonsági rések észlelése lokális, vagy távoli gépeken történik

8.2 Snort

A Snort egy behatolás jelző rendszer, ami képes azonnali forgalomelemzésre és csomagok naplózására IP hálózatokban. Protokollok analizálására is használható, valamint az adatforgalomban is kereshetünk vele.

---

[1] Az "Ad hoc" lekérdezéseket a rendszerek használata során, maguk a felhasználók teszik, célirányos feladatok ellátására. Ezek a lekérdezések a rendszerekbe nem mint beépített funkciók jelentkeznek

[2] Szó szerint vizsgálat. Általában egy működő rendszerre, folyamatra, termékre vonatkozik, megvizsgálva, hogy az mennyire felel meg az elvárásoknak, előírásoknak

A hálózati forgalmat előre elkészített szabályok alapján figyeli és elemzi. A Snort a harmadik, és az a fölötti rétegekben működő rendszer.

3 fő funkciója van

1. hálózati csomagok elfogása
2. csomagok naplózása, ami hasznos a hálózati forgalom elemzésénél
3. behatolás-érzékelő rendszer

Tulajdonságok:

- különböző támadások és betörési próbálkozások észlelésére képes (puffer túlcsordulás, rejtett port szkennelés, SMB[1] vizsgálat, CGI[2] támadások, OS újlenyomatok kitalálásának kísérletét is jelzi)
- egy rugalmas szabály alapú nyelvet használ arra, hogy szabályozza, milyen forgalmat gyűjt, és milyet enged át
- a detektáló motor moduláris felépítésű, így könnyen továbbfejleszthető, kiegészíthető
- valós időben tud figyelmeztetéseket küldeni, amik a syslogban, a felhasználó által megadott fájlban, vagy felugró ablakban is megjelenhetnek

8.3 Nmap

Az Nmap egy ingyenes, nyílt forráskódú eszköz, hálózatok felderítésére és biztonsági auditálásra. Az Nmap raw IP csomagokat használ újszerű módon, hogy felfedezze milyen hosztok érhetőek el a hálózaton, milyen szolgáltatásokat nyújtanak, milyen operációs rendszert futtatnak és milyen csomagszűrést, vagy tűzfalat használnak. Nagy hálózatokra tervezték, de egy hoszt tesztelésével is elboldogul. Az Nmap a legtöbb operációs rendszeren futtatható, konzolos és grafikus felülettel is rendelkezik.

Főbb tulajdonságai:

- *rugalmas*: sok technikát alkalmaz a hálózatok felderítésére, amikben lehetnek IP szűrők, tűzfalak, routerek stb. Ebbe beletartozik a sokféle portvizsgálati mechanizmus (TCP, UDP), OS detektálás, verziódetektálás, pingelési módszerek.
- *erőteljes*: nagyon nagy hálózatok vizsgálatára is alkalmas.
- *hordozható*: a legtöbb operációs rendszert támogatja. Linux, Windows, FreeBSD, Solaris, OSX…

[1] Az *SMB* (*Server Message Block*) egy állomány- és nyomtatómegosztást lehetővé tevő protokoll.

[2] A CGI (Common Gateway Interface) scriptek, vagy kis programok segítségével dinamikussá tehetők a weblapok.

- *könnyű használat*: annak ellenére, hogy mennyi minden valósítható meg vele, könnyen kezelhető. Parancssorból, és grafikusan is irányítható.
- *ingyenes*: a fő célja az Nmap fejlesztésének, hogy segítse az internet biztonságosabbá tételét, és hogy a rendszergazdáknak, auditoroknak egy olyan eszközt biztosítson, amivel feltérképezhetik a hálózatukat.
- *jól dokumentált*: sok folyamatosan frissülő felhasználói dokumentáció, leírás és használati útmutató érhető el hozzá.
- *jól támogatott*: segítőkész közösség áll körülötte, fejlesztők és felhasználók. Különböző levelező listák is segítenek, ha problémába ütközünk.

8.4 Wireshark

A wireshark a világon a legkedveltebb hálózati protokollanalizáló program. Nagyon sok helyen használják, többek közt az iparban és az oktatásban is. Ez egy ingyenesen elérhető szoftver, amivel képesek vagyunk a hálózaton átmenő forgalom figyelésére. Több rétegben használhatjuk a csomagok elemzésére. A hálózati forgalom figyelésére azért van szükség, hogy diagnosztizáljuk az esetleges hálózati hibákat, felfedezzük a biztonsági réseket, optimalizálni tudjuk a hálózati forgalmat és megtudjuk, hogy egy program milyen hálózati forgalmat generál.

Fő tulajdonságai:

- több száz protokollt ismer fel és ez a szám folyamatosan bővül
- az elfogott csomagokat offline módban is kielemezhetjük
- több platformra is elérhető: Windows, Linux, OSX, Solaris, FreeBSD…
- az elfogott hálózati adatokat elemezhetjük grafikus felületen, vagy karakteres módban is egy konzol segítségével
- VoIP[1] analizálásra is alkalmas
- más hasonló szoftverek által elfogott adatokat is fel tudja dolgozni (tcpdump, Cisco Secure IDS iplog, NetXray, Novell LANalyzer…)
- az elfogott adatokhoz tartozó fájlokat gzip-be tudja tömöríteni
- adatokat tud olvasni különböző eszközökről (Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, FDDI)

[1] Az Internet Protokoll feletti beszédátvitel – elterjedt nevén VoIP (Voice over IP), vagy IP-telefónia – a távközlés egy olyan formája, ahol a beszélgetés nem a hagyományos telefonhálózaton, hanem az Interneten vagy más, szintén IP-alapú hálózaton folyik

- az elfogott adatokat szűrhetjük annak megfelelően, hogy milyen információra van szükségünk
- a kimeneteket XML, PostScript, CVS, vagy szöveges formába is exportálhatjuk

Felhasznált irodalom:

Bevezetés

http://en.wikipedia.org/wiki/Network_security

Behatolás-megelőzés

http://en.wikipedia.org/wiki/Intrusion-prevention_system

Behatolás-érzékelés

http://en.wikipedia.org/wiki/Intrusion-detection_system

Behatolások kezelése

http://csrc.nist.gov/publications/nistpubs/800-61-rev1/SP800-61rev1.pdf

Hálózat felderítése

http://www.cs.cornell.edu/boom/1999sp/projects/Network%20Topology/topology.htm

Hálózati adatforgalom analizálása:

http://www.securityfocus.com/infocus/1696

Tűzfalak:

http://www.howstuffworks.com/firewall.htm

http://www.vicomsoft.com/knowledge/reference/firewalls1.html

DMZ :

http://en.wikipedia.org/wiki/Demilitarized_zone_(computing)

Eszközök:

Nessus: http://nessus.org

Snort: http://www.snort.org

Nmap: http://nmap.org/

Wireshark: http://www.wireshark.org/