ELOSZTOTT FENYEGETETTSÉG FELMÉRÉS

Leitold Ferenc
Dunaújvárosi Egyetem, Secudit Kft.
fleitold@secudit.com

Hadarics Kálmán
Dunaújvárosi Egyetem
hadarics@uniduna.hu

DOI: 10.31915/NWS.2018.2

Distributed Vulnerability Assessment Electronic information systems are used in nearly every area of life today. Besides computers smart and IoT devices turn up. However, when IT systems are used online there are cyber-threats too. The so called cyber criminals can steal unauthorised data and credentials by means of malicious codes or can have a harmful effect on IT security. If we want to observe the protection of an IT system and infrastructure against threats we must consider several relevant relating parameters. Three factors are identified in the applied model of cyber-threats – Distributed Vulnerability Assessment (DVA):

1. characteristics and prevalence of harmful cyber-threats;
2. vulnerabilities of IT infrastructure and its processes;
3. vulnerabilities deriving from users' behaviour.

Using a metric, the impact of a threat typical of a given infrastructure can be determined with a mathematical model. This metric means the probability of at least one threat attacking successfully at least one device in the IT infrastructure used by the given users. All available information must be considered in the case of the three cornerstones for the operation of the model. Such information is the prevalence, the necessary hardware and software elements or the demanded user activity. In the case of user behaviour, the most important characteristic is when and how the user uses the IT devices, to what extent he tends to open e-mail attachments or visit unknown web sites. In the case of IT infrastructure what hardware or software elements are present or absent and how they affect the operation of the observed harmful code. This, obviously, relates to the protection systems installed on the devices of the IT infrastructure.

Using our mathematical approach, the integrated vulnerability is decomposed and distributed to the contributing elements of individual user susceptibility, individual IT infrastructure elements, and the individual protecting cybersecurity services and applications. From the DVA results, vulnerability is quantitatively attributed to the various internal contributing components (e.g., user identities, ports, protocols, protection layers). This allows different contributing components to be assessed using comparable metrics (e.g., user security awareness vs. infrastructure patch condition vs. efficacy of anti-malware). DVA allows information security managers to pose and compare the results of „what if" queries to see the vulnerability reduction of various available options that might not otherwise be quantitatively comparable (e.g., investment in employee security awareness programs vs. hardening IT infrastructure vs. adding additional cybersecurity applications and services. The framework, formulae, and relevant examples of applying DVA to single LAN and multiple LAN enterprise networks are described.

This paper describes our model capable of determining the metric of threats. The paper includes the applied mathematical formulae to present the practical application of the model.
Keywords: cyber security, DVA, vulnerability metric, threat

Bevezetés

A DVA (Distributed Vulnerability Assessment) technológia a Dunaújvárosi Egyetem és a Secudit közös kutatási munkája alapján jött létre. A DVA részletes leírást ad egy szervezet internetes támadási sebezhetőségeiről. A módszer szerint első lépésként az egyedi felhasználók és az informatikai infrastruktúra elemeinek sebezhetőségét az egyes ismert fenyegetésekre vonatkozóan kell felmérni, majd ezeket az eredményeket kombinálni az adott szervezet számára releváns fenyegetésekre vonatkozóan. A módszer egy adott szervezet integrált kiber-támadási sebezhetőségét a jelenleg ismert fenyegetések elterjedtségét és hatékonyságát; a felhasználók biztonságtudatos viselkedését; és az informatikai infrastruktúra gyengeségeit alapul véve értékeli. Matematikai módszereket alkalmazva az integrált sebezhetőség felbontható arra, hogy az egyes felhasználók, illetve az egyes IT infrastruktúra elemek milyen mértékben járulnak hozzá az integrált sebezhetőséghez, a teljes szervezet fenyegetettségéhez. A DVA-eredményekből a fenyegetettség mennyiségi szempontból hozzárendelhet a különböző belső hozzájáruló összetevőkhöz (például felhasználói azonosító, portok, protokollok, védelmi rétegek). Ez lehetővé teszi, hogy különböző közreműködő komponenseket összehasonlítható mérőszámokkal értékeljünk (pl. felhasználói biztonságossági tudatosság, az infrastruktúra javításának lehetősége, illetve a rosszindulatú programok elleni védelem hatékonysága alapján). A DVA lehetővé teszi az információbiztonsági menedzserek számára, hogy a „mi lenne, ha" típusú lekérdezések eredményei alapján összehasonlíthassák a különböző rendelkezésre álló lehetőségeket a szervezet fenyegetettségének csökkentése érdekében, amelyek egyébként nem lennének mennyiségi szempontból összehasonlíthatók (pl. további cybersecurity alkalmazások és szolgáltatások.)

1. Fenyegetettségek modellezése

Ahhoz, hogy egy kártékony támadás sikeres legyen egy védett hálózattal szemben, a kártékony kód sikeres végrehajtása szükséges. A felhasználói oldalon a legegyszerűbb minimális viselkedés nem más, mint a végpont eszköznek az internethez történő csatlakoztatása. Az informatikai biztonsági metrikák manapság a védett IT-re (pl. folyamatos sérülékenység-tesztelés), illetve a kártevők tevékenységére, tulajdonságaira (pl. védelmi rendszerek tesztelése) [6] fókuszálnak. A felhasználói magatartásra vonatkozó informatikai biztonsági metrika kevésbé fejlett [3], habár a hálózati forgalom megfigyelése lehetőséget ad a fejlesztésükre (pl. NetFlow/IPFIX). A passzív figyelés mellett az interaktív metrikát is alkalmazhatjuk [10].

A sikeres kártékony támadásokat a védett környezetben megvalósítható kártékony tevékenység és a megfelelő felhasználói magatartás metszeteként lehet reprezentálni. Ez a koncepcionális keret az NSS Lab által használt működési szabályokra épül [18], ugyanakkor praktikus és kényelmes egyszerűsítése a támadási felületek komplett kezelésének. Az alábbiakban csak a humán-interaktív végpontokra

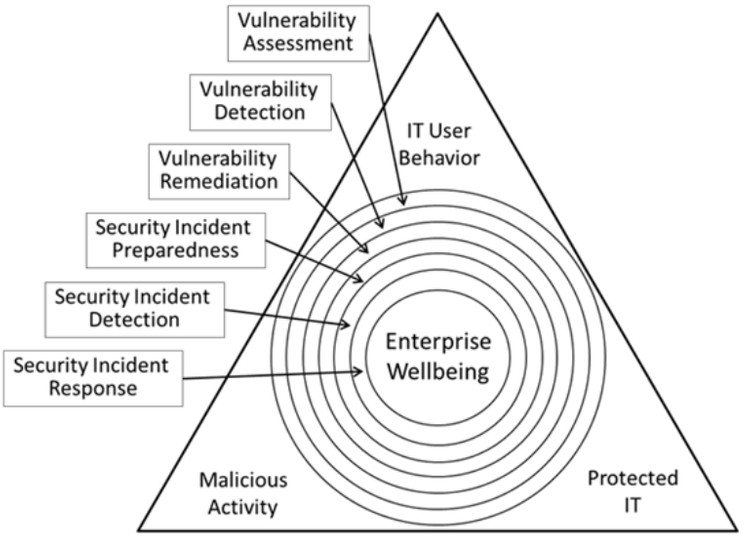
fókuszálunk (IT), a beágyazott rendszerek biztonsági architektúrájával (IoT, OT) jelenleg nem foglalkozunk. Három különálló, de erősen interaktív sérülékenységi forrást veszünk figyelembe:

1. kártékony tevékenység azok által, akik saját céljaikra használják ki a hálózat képességeit, hogy megsértsék a megbízható IT rendszer védelmét;
2. veszélyes IT felhasználói magatartás (pl. alkalmazottak, vevők, beszállítók); és
3. védelem nélküli sérülékenység az IT hálózati infrastruktúrában.

A legkritikusabb sérülékenység e három terület közös részében, metszetében található (1. ábra). E sérülékenységek megfelelő láthatóságot, ellenőrzést és megkülönböztetést követelnek a megfigyelésükhöz, megértésükhöz és az ellenük történő hatékony védekezéshez. A meglevő és esetleg felmerülő sérülékenységek láthatóságához éber kockázatelemzés szükséges, ami mindhárom területet figyeli (1. ábra).

1. ábra Az IT sérülékenység komponenseit és tényezőit három területre lehet osztani, melyek mindegyikének saját módszere és eszközei vannak a láthatóság, ellenőrzés és megkülönböztetés céljára [13]

Az információs folyamatok sérülékenységeinek láthatósága szükséges, de önmagában elégtelen az informatikai biztonság szempontjából. A sérülékenység értékelése a biztonság biztosításának legkülsőbb rétege. A következő rétegek: sérülékenység érzékelés, sérülékenység javítása, biztonsági incidensre való felkészülés, biztonsági incidens érzékelése, és biztonsági incidensre való reagálás (2. ábra).



2. ábra Sérülékenység felmérése a teljes biztonság érdekében a szervezet jóléte céljából [13]

A szervezet jólétének biztosításához a sérülékenységek kezelése a sérülékenységek forrásainak gyakorlati és hatékony azonosítását követeli meg. A biztonsági incidensre való reagálás követelményét az esemény információkezelő rendszerek elégítik ki (SIEM). A sérülékenységek hatékony kezeléséhez az informatikai sérülékenység hármas modellje szükséges. A korábbi szabályokból eredően [11, 12] a hármas modell a sérülékenység mérését 3 forrásra osztja: i) kártékony tevékenység; ii) védelemmel rendelkező IT; és iii) nem megfelelő felhasználói magatartás. Mindegyik forrásban specifikus tényezőket azonosítunk és jellemzünk (pl. vírusküldés és kihasználás a kártékony tevékenységi hármasban). A modell alapot ad a tényezők korrelációjához és kombinálásához a sérülékenységek integrált nézetéhez.

2. A sérülékenység forrásainak korrelációja és kombinálása

Definiáljuk a következőket:

L: minden elérhető fenyegetési helyszín halmaza (pl.: világ, Európa, USA, Magyarország, …)

T_{all}: minden lehetséges kártevő halmaza

(megjegyzés: csak a fenyegetések al-halmazára fókuszálunk, csak a programozott támadásokat nézzük)

T_1: minden lehetséges belső kártevő halmaza, $l \in L, T_l \subset T_{all}$

U: minden felhasználó halmaza

I: minden lehetséges eszköz halmaza

P: minden elérhető védelem halmaza

UT: minden lehetséges T-ben levő kártevő által használt felhasználói trükk halmaza

A sérülékenység három forrásra vezethető vissza (támadói találékonyság, infrastruktúra gyengesége, nem megfelelő felhasználói magatartás). Minden adott kártevőre vagy kártevő osztályra a következőket tudjuk megbecsülni:

1. A szervezettel szemben a támadó által használt kártevő tevékenység valószínűsége (p_{prev}):

$$p_{prev}(t,l) = \frac{number\ of\ computers\ infected\ by\ \boldsymbol{t}\ inside\ \boldsymbol{l}}{number\ of\ all\ computers\ inside\ l}$$

ahol l $t \in T_l$ és $l \in$

2. A szervezet által lehetővé tett sikeres támadás valószínűsége (p_{device}):

$$p_{prot}\ (t,p) = \frac{number\ of\ successful\ attempts\ of\ \boldsymbol{t}\ thru\ the\ protection\ \boldsymbol{p}}{number\ of\ all\ attempts\ of\ \boldsymbol{t}\ thru\ the\ protection\ \boldsymbol{p}}$$

ahol $t \in T_l, l \in L$ és $p \in P$;

$$p_{device-prot}(t,i) = \min_{for\ all\ p\ protecting\ i} p_{prot}(t,p)$$

ahol $t \in T_l, l \in L$ és $i \in I$;

$$p_{device-elements}(t,i) = \begin{cases} 1, if\ t\ can\ work\ on\ i \\ 0, if\ t\ cannot\ work\ on\ i \end{cases}$$

ahol $t \in T_l, l \in L$ és $i \in I$;

$$p_{device}(t,i) = p_{device-elements}(t,i) \cdot p_{device-prot}\ (t,i)$$

ahol $t \in T_l, l \in L$ és $i \in I$;

3. A felhasználók által lehetővé tett sikeres támadás valószínűsége ($p_{usertrick}$, p_{user}, p_{usage}):

$$p_{usertrick}(t,ut) = \frac{number\ of\ attempts\ of\ \boldsymbol{t}\ where\ t\ used\ \boldsymbol{ut}}{number\ of\ all\ attempts\ of\ \boldsymbol{t}}$$

ahol $t \in T_l, l \in L, ut \in UT$

$$p_{user-usertrick}(u, ut) = \frac{number\ of\ successful\ attempts\ of\ \boldsymbol{ut}\ on\ \boldsymbol{u}}{number\ of\ all\ attempts\ of\ \boldsymbol{ut}\ on\ \boldsymbol{u}}$$

ahol $u \in U, ut \in UT$

$$p_{user}(u, t) = 1 - \prod_{for\ all\ ut\ used\ by\ t} (1 - p_{usertrick}(t, ut) \cdot p_{user-usertrick}(u, ut))$$

ahol $u \in U, t \in T_l, l \in L, ut \in UT$

A 3 fő bemeneti osztályt (p_{prev}, p_{device}, p_{user}) kombinálni lehet, hogy a sikeres támadás teljes valószínűségét megtudjuk (feltéve, ha a támadás, felhasználó és az IT infrastruktúra komponensét is figyelembe vesszük):

$$p_s\ (l) = 1 - \prod_{for\ all\ t,u\ and\ i} (1 - p_{user}(t, u) \cdot p_{device}(t, i) \cdot p_{prev}(t, l))$$

ahol $u \in U, i \in I, t \in T_l, l \in L$

A fenti számításnál azonban még nem tettünk különbséget aközött, hogy egy felhasználó milyen intenzitással használja a számítógépét. Nyilvánvaló különbség ha egy felhasználó naponta 10 percet használja a számítógépét vagy ha 10 órát használja a számítógépét. A számítógépek használatára vonatkozó intenzítást az alábbiak szerint kalkulálhatjuk:

Legyen

$$\mu(t, u) = \frac{number\ of\ attempts\ of\ \boldsymbol{t}\ are\ enabled\ by\ the\ user\ \boldsymbol{u}}{number\ of\ attempts\ of\ \boldsymbol{t}\ are\ enabled\ by\ the\ average\ user}$$

Ekkor

$$p_s\,(l) = 1 - \prod_{for\ all\ t,u\ and\ i} (1 - p_{user}(t,u) \cdot p_{device}(t,i) \cdot p_{prev}(t,l))^{\,k(t,u)}$$

ahol $u \in U, i \in I, t \in T_l, l \in L$ *és*

$$k(t,u) = \frac{T}{\Delta T} \cdot \frac{T_u}{T_{average}} \cdot \mu(t,u)$$

ahol

ΔT az elterjedtségekre (prevalence) vonatkozó időintervallum,

T az az időintervallum, amire az általunk számított valószínűségi mérték vonatkozik,

T_u az az időintervallum, amennyi ideig az u felhasználó használja a számítógépet,

$T_{average}$ az az időintervallum, amennyi ideig egy átlagos felhasználó használja a számítógépet,

μ(t,u) a fentiek alapján számított érték.

A fentiek alapján a sikeres támadás külön mért kombinált valószínűségeit össze lehet vetni és sorrendbe állítani (p_{s1}, p_{s2}, p_{s3}, …). Tehát egy azonosított magas prioritású sérülékenységet (p_{si}) le lehet bontani alkotó sérülékenységi forrásaira (p_{ai}, p_{bi}, p_{ci}) lehetővé téve a javítást ott, ahol az leginkább lehetséges.

3. Összegzés

A fentiekben módszert mutattunk be a sérülékenység mérésére. Három információforrást használunk: külső informatikai fenyegetés intelligencia („biztonsági intelligencia”), szervezeti IT infrastruktúra gyengeség („behatolás tesztelés”), és a felhasználók fogékonysága, naivsága a támadásokra („felhasználói magatartás”). A módszer lehetővé teszi a mért források kombinálását egy metrikába, amit összevethető sérülékenységekre bonthatunk. A módszer számszerűsíti a relatív sérülékenység evolúcióját időben, külön mérheti az egyedi osztályok (LAN) sérülékenységét és a specifikus fenyegetéseket (pl. zsaroló vírusok, adathalászat). A módszer előrejelzi a potenciális javítási tevékenység következményeit („Mi lesz, ha?”), ezáltal segíti a biztonsággal kapcsolatos döntéshozatalt az adott helyzetben.

A fenyegetettségi mérték meghatározása és folyamatos monitorozása az információbiztonság fenntartása, illetve szintjének javítása érdekében lehetőséget ad

- a legkevésbé biztonságtudatos felhasználók azonosítására;
- a fenyegetettségi mértéket leginkább meghatározó hardver, illetve szoftverelemek azonosítására;
- a fenyegetettségi mérték meghatározására a szervezet különböző részlegeire vonatkozóan, illetve ezek összehasonlítására;

– a biztonsági szint növelését célzó intézkedések hatásának elemzésére (pl. mennyire javul a fenyegetettségi mérték, ha az összes számítógépen Windows 10-re cseréljük az operációs rendszert, vagy ha a legkevésbé biztonságtudatos 10 felhasználót információbiztonsági oktatásra küldjük).

A programozott fenyegetések száma manapság 7-800 millió körüli, az aktív támadások köre folyamatosan változik, ráadásul a támadások kb. 90%-át egyedi fertőzések okozzák. Ilyen körülmények között az egy szervezetre vonatkozó veszélyeztetettség mérése sokkal inkább becslés, mint pontos számítás. A bemeneti adatok minél pontosabb meghatározásával, a figyelembe vett kártevők körének kiválasztásával pontosabbá tehető az analízis.
A cikkben leírt számítás annál használhatóbb továbbá, minél gyorsabban tudja követni a valós eseményeket. Az infrastruktúra, illetve a felhasználói viselkedés felmérésére, monitorozására léteznek automatikus módszerek, melyek valós időben tudnak adatot szolgáltatni a számtásokhoz. A fenyegetések oldaláról viszont a bemeneti adatok csak valamilyen késleltetéssel állnak rendelkezésre, amit persze több módszerrel lehet javítani. A másik oldalról viszont egy-egy sikeres fertőzés átlagosan hónapokat is egy szervezet rendszerében lehet, mielőtt az felderítésre kerülne, így az ismertetett módszerrel mindenképpen hatékonyabbá tehető a védekezés.

Irodalomjegyzék

[1] ARROTT, A., F. Lalonde Levesque, D. Batchelder, and J.M. Fernandez. „Citizen cyber-security health metrics for Windows computers". Proceedings of Eastern European eGov Days Conference, EEGOV, Budapest, Hungary. 2016.

[2] BATCHELDER, D., et al. „Microsoft Security Intelligence Report." Volume 18: July-December 2014, Microsoft, 2015.

[3] CHAPMAN, M.T., „Establishing metrics to manage the human layer." ISSA Security Education Awareness Special Interest Group, 2013.

[4] CLEMENTI, Andreas, Peter Stelzhammer, and Fernando C. Colon Osorio. „Global and local prevalence weighting of missed attack sample impacts for endpoint security product comparative detection testing." Malicious and Unwanted Software: The Americas (MALWARE), 2014 9th International Conference on. IEEE, 2014.

[5] COLON OSORIO, F.C., and A. Arrott. „Fabric of security - changing our theory and expectations of modern security". Proceedings of Eastern European eGov Days Conference, EEGOV, Budapest, Hungary. 2016.

[6] EDWARDS, S.E., R. Ford, and G. Szappanos., „Effectively testing APT defenses". Virus Bulletin Conference, Prague, Czech Republic, 2015.

[7] KLEINER, A., P. Nicholas, K. Sullivan, „Linking Cybersecurity Policy and Performance, Microsoft Trustworthy Computing", 2013,

[8] KSHETRI, Nir. „Cybercrime and Cybersecurity in the Middle East and North African Economies." Cybercrime and Cybersecurity in the Global South. Palgrave Macmillan UK, 2013.

[9] LALONDE LEVESQUE, F., A. Somayaji, D. Batchelder, and J.M. Fernandez. „Measuring the health of antivirus ecosystems." Malicious and Unwanted Software (MALWARE), 2015 10th International Conference on. IEEE, 2015.

[10] LALONDE LEVESQUE, F., J. M. Fernandez, and A. Somayaji. „Risk prediction of malware victimization based on user behavior." Malicious and Unwanted Software: The Americas (MALWARE), 2014 9th International Conference on. IEEE, 2014.

[11] LEITOLD, F and K. Hadarics. „Measuring security risk in the cloud-enabled enterprise." Malicious and Unwanted Software (MALWARE), 7th International Conference on Malicious and Unwanted Software, pp: 62-66, ISBN: 978-1-4673-4880-5. 2012.

[12] LEITOLD, F. „Security Risk analysis using Markov Chain Model." 19th Annual EICAR Conference, Paris, France. 2010.

[13] LEITOLD, F., A. ARROTT and K. HADARICS, „Quantifying cyber-threat vulnerability by combining threat intelligence, IT infrastructure weakness, and user susceptibility" 24th Annual EICAR Conference, Nuremberg, Germany, 2016

[14] LEITOLD, F., A. ARROTT and K. HADARICS, „Automating visibility into user behavior vulnerabilities to malware attack" Proceedings of the 26th Virus Bulletin International Conference (VB2016), pp. 16-24, Denver, USA, 2016.

[15] MICROSOFT. „Evolution of malware and the threat landscape - a 10-year review". 2012.

[16] MICROSOFT. „Malicious Software Removal Tool (MSRT) „. Microsoft Knowledge Base, article KB890830 revision 161.2, https://support.microsoft.com/en-us/kb/890830

[17] RUBENKING N., "Why Microsoft Doesn't Need Independent Antivirus Lab Tests". PC Magazine, 28 October 2013.

[18] SHAH P, Phatak V, Scipioni R, inventors. „Adaptive intrusion detection system." United States patent application US 10/443,568. 2003 May 22.