Kompüter şəbəkələrində zərərli proqramların analizi üsulları

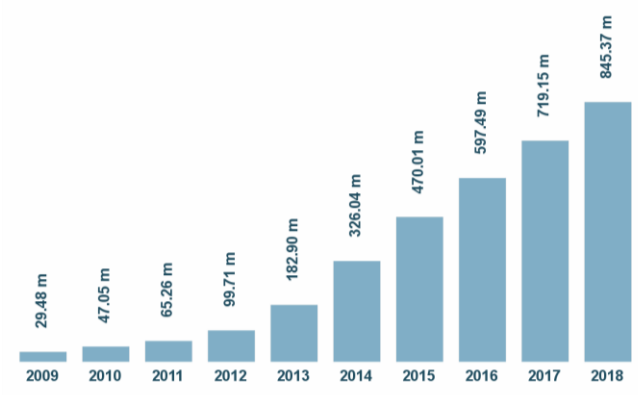
Tural Yunusov
AMEA İnformasiya Texnologiyaları İnstitutu, Bakı, Azərbaycan
turaly@mail.ru

***Xülasə* — Məqalədə kompüter şəbəkələrinin təhlükəsizliyi, zərərli proqramlar haqqında informasiya verilmişdir, zərərli proqramların analizi üsulları, statik analiz üsulu, dinamik analiz üsulu, zərərli proqramların virtual mühitdə analizi, zərərli proqramların yoluxma üsulları tədqiq olunmuşdur.**

Açar sözlər – kompüter şəbəkələri, zərərli proqramlar, rootkit, zərərli mobil kod, kuki faylları, hooking, sandbox.

I. GİRİŞ

Zərərli proqram dedikdə, məlumatın məhv edilməsi, oğurlanması, proqram və ya əməliyyat sisteminin konfidensiallığına, tamlığına və əlçatanlığına ziyan vurmaq (viruslar, Troya atları, Zərərli mobil kodlardan istifadə) məqsədi ilə hazırlanmış zərərli kodlar nəzərdə tutulur. Xoşagəlməz proqram çərçivəsində nəzər yetirilən virus, soxulcan, troya atları, casus və reklam tərkibli proqramlar illərdir fərdi və korporativ kompüter şəbəkələrinə təhdit olmuşlardır. Kompüter istifadəçilərinin sayı artdıqca, komptüter şəbəkələrinin müxtəlif təhdidlərə qarşı həssas vəziyyətə gəlməsi də bir reallıqdır [1].



Şəkil 1. Zərərli proqramların illər üzrə yayılma statistikası

AV–TEST İT təhlükəsizliyi institunun verdiyi statistikaya əsasən 2018–ci ildə zərərli proqramların yayılması digər illərə görə daha çoxdur (Şək.1).

II. ZƏRƏRLİ PROQRAMLAR

Zərərli proqramları aşağıdakı kimi klasifikasiya etmək olar.

Soxulcan – müstəqil işləyə biləcək və digər kompüter şəbəkələrində olan sistemlərə kopyalana bilən kiçik proqramlara verilən ad. Əlavə olaraq botnet qurmaq üçün də istifadə olunur.

Virus – müstəqil formada işləyə bilmir, aktivləşdirilməli və əməliyyat sistemi fayllarına əlavə edilməli bir kod hissəsi kimi müəyyən edilir.

Trojan – və ya troya atları, əsasən faydalı proqramlara əlavə edilən zərərli kodun gizli hissəsidir. Arxa tərəfdə port açma və məsafədən idarəetmə kimi funksiyaları istifadəçidən xəbərsiz yerinə yetirir.

Spyware – casus proqramlar olaraq adlandırılan bu proqramlar istifadəçi kompüterindəki həssas informasiyaları (istifadəçi informasiyalarını, şifrələri, şəxsi məlumatları) yığan və hücumçunun kompüterinə göndərən proqramlardır.

Bot – istifadəçi kompüterində gizli quraşdırılan və yoluxmuş kompüterin resurslarından istifadəçinin xəbəri olmadan istifadə etməklə, müəyyən əməlləri yerinə yetirməyə imkan verən uzaqdan idarə oluna bilən zərərli proqramdır. Bu sistemlər əsasən spam göndərmə, DDOS hücum etmə kimi işlərdə istifadə edilir.

Rootkit – hücuma məruz qalan sistem üzərində edilən bütün əməliyyatları gizləyən (aktiv əməliyyatlar, fayllar, qeydlər və.s) proqramlardır. Kompüter yüklənməsi zamanı yoluxan və öz fayl sistemi olan yeni nəsil zərərli proqramlara bootkit adı verilmişdir.

Zərərli mobil kod – mobil kod çox vaxt istifadəçinin təlimatı olmadan yerli sistemdə icra edilmək üçün uzaq sistemdən ötürülən proqram təminatıdır. Zərərli mobil kod faylları yoluxdurmaması və ya özünü təkrarlamaması ilə virus və soxulcanlardan fərqlənir. Mobil kod üçün populyar anlayışlara Java, ActiveX, JavaScript və VBScript daxildir.

İzləyici Kuki faylları – Kuki xüsusi vebsaytdan istifadə barədə məlumatı saxlayan kiçik “data” faylıdır. Sessiya kukiləri yalnız bir vebsayt sessiyasında etibarlı olan müvəqqəti kukilərdir. Daimi kukilər kompüterdə həmişəlik saxlanır ki, sayta tez–tez daxil olan istifadəçiləri müəyyən edə bilsin. Daimi kukilər saytın istifadəçinin daxil olmasını və ya gələcək daxilolmalar zamanı davranışı avtomatik fərdiləşdirməsi üçün bir vebsaytda istifadəçi göstəricisini qeydə almaq üçün nəzərdə tutulmuşdur. Bu yolla daimi kukilər vebsaytlara istifadəçilərinə daha səmərəli xidmət göstərməyə kömək edir. Daimi kukilər istifadəçinin xəbəri və ya razılığı olmadan şübhəli məqsədlərlə istifadəçinin veb axtarışlarını izləmək üçün casus proqram kimi

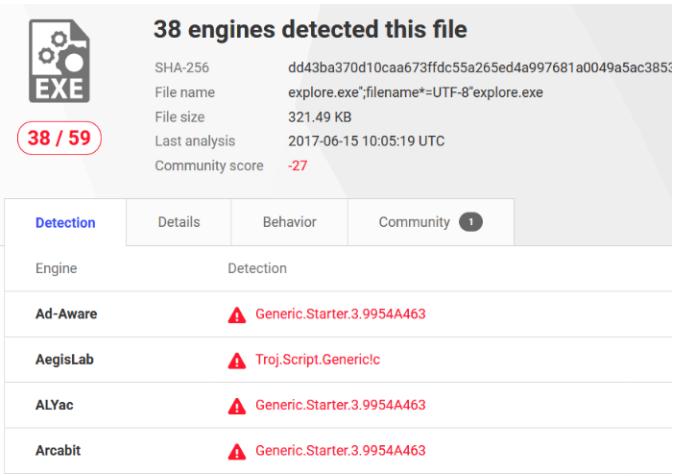
DOI: 10.25045/NCInfoSec.2018.24

də istifadə edilə bilər. Məsələn, marketinq firması bir çox veb saytlarda reklam yerləşdirə və istifadəçinin cihazında olan bir kukidən istifadə edərkən onun bütün veb saytlardakı fəaliyyətini izləyə və istifadəçinin davranışından ətraflı məlumat toplaya bilər. Bu cür istifadə edilən kukilər izləyici kukilər kimi tanınır. İzləyici kukilər tərəfindən toplanan məlumat çox vaxt digər tərəflərə satılır və istifadəçilərə reklam və başqa məzmun göndərmək üçün istifadə edilir.

III. ZİYANKAR PROQRAMLARIN ANALİZ ÜSULLARI

Antivirus proqramları əsasən zərərli proqramların müəyyən bir hissəsindən yaradılan imzaları (signature) verilənlər bazasında saxlayıb, buradan müqayisə edərək zərərli proqramı aşkarlaya bilirlər [2]. Günümüzdə bu iş üçün istifadə edilən ən yaxşı alət VirusTotal–dır.

VirusTotal fayl analizinə pulsuz icazə verən veb saytdır. Təqribən 56 antivirus sistemini özündə birləşdirir. Faylları həm veb üzərindən, həm də e-poçt vasitəsilə göndərmək olur. Sistem yalnız ona göndərilən kiçik ölçüdəki faylları analiz edir, kompüteri analiz etmir (Şək.2).



Şəkil 2. VirusTotal vasitəsilə zərərli proqramın aşkarlanması

Bu üsul çox vaxt müvəffəqiyyətlidir, lakin günümüzdə inkişaf etmiş zərərli proqramları bu üsulla aşkarlamaq mümkün olmur. Çünki zərərli proqram tez–tez təkminləşdirilir və bu vəziyyət imzanın dəyişməsinə səbəb olur. Fərqli üsullarla (şifrləmə, başqa dildə kodlaşdırmaq, kodları başa düşülməyən formaya gətirmək) imza dəyişikliyinə səbəb olurlar [3].

Bu səbəblə zərərli proqramları aşkarlamaq və analiz etmək üçün aşağıdakı üsullardan istifadə edilir.

Statik analiz üsulu zərərli proqramı aktivləşdirmədən bir başa, kod səviyyəsində hansı xidmətlərə giriş təmin etdiyinin və ya hansı dəyişikliklərin apardığının analiz edilməsidir.

Dinamik analiz isə təcrid edilmiş virtual mühitdə zərərli proqramı aktivləşdirmək və etdiyi dəyişiklikləri (daimi sistem qeydləri, fayl sistemi, prosesor əmrləri) izləyərək analiz etmə üsuludur.

A) Statik analiz üsulu

Əgər mənbə kod (source code) əlimizdədirsə statik analiz üsulu ilə bütün funksiyaları test edə bilərik. Əlimizdə sadəcə binar kod (bir və sıfırdan ibarət olan maşın kodu) mövcuddursa verilənlərin tipi kimi informasiyalar itəcəyindən analiz çətinləşəcəkdir. Hex (onaltılıq say sistemi) dəyişdirmə proqramları vasitəsi ilə xəta izləmə xüsusiyyətlərini də istifadə edərək və müəyyən əmrlər sətri axtarılaraq zərərli proqramın işləmə prinsipləri haqqında məlumat almaq mümkündür. Tərs mühəndislik (Reverse Engineering) metodları ilə əlimizdə olan binar kod (binary code) zərərli proqram nümunəsindən üst səviyyə kodlaşdırılmış mənbə kod (source code) əldə etmək mümkündür. Bu kodu nə qədər müvəffəqiyyətli geri kodlaşdıra bilsək, alınan kodun işləməsi də insan gözüylə aydınlaşdırılması o qədər asanlaşacaqdır [4].

B) Dinamik analiz üsulu

Dinamik analiz üsulunda zərərli proqramların analizi aşağıdakı formada yerinə yetirilir.

Funksiya çağırışlarının monitorinqi (Function Call Monitoring) – zərərli proqram aktivləşdirildiyi vaxt, əgər funksiya çağırma (bu əməliyyat sisteminə aid istifadəçi funksiyası ola bilər) prosesi yerinə yetirirsə bu mərhələdə ortaya girərək bəzi məlumatlar əldə edə bilərik. Bu proses “hooking” adlanır. Hooking funksiyası, bir qeydiyyat faylına, bu prosesin adını, giriş parametrləri kimi məlumatları əlavə edir [5].

Əməliyyat sistemləri fundamental funskiyaları olan fayl yaratma, şəbəkəyə sorğu göndərmə, avadanlıq üzərində kod aktivləşdirmə kimi proseslərə əməliyyat sistemləri üzərində işləyən proqramlara birbaşa giriş icazəsi verə bilmir bunu etmək üçün bir interfeys (APİ) üzərində bu prosesi nəzarətlə verir. Windows APİ – ləri şəbəkə, təhlükəsizlik, sistem xidmətləri və idarəçiliyi kateqoriyaları altında toplanır.

Şəkil 3. Hooking prosesi

Zərərli proqramların mənbə kodu məlumdursa bu koda hooking funksiyaları əlavə edilə bilir (Şək.3). Əlimizdə ancaq binar kod varsa binar kodunu təkrar yazma (binary rewriting) üsulu ilə maşın kodu üzərində dəyişiklik aparılaraq Hex dəyişdirici proqramlarında prosesin olduğu yaddaş ünvanını göstərən rəqəmi hook funksiya ünvanı ilə dəyişdirib hooking prosesi başa çatdıqdan sonra real proses ünvanına yönləndirilə bilər.

Yaddaş üzərində dəyişdirmə prosesinə inyeksiya deyilir. Kodun istifadə etdiyi kitabxanalar “stub” adı verilən, daha sonra əsliylə dəyişdiriləcək müvəqqəti kod hissələri ilə dəyişdirilir. Aşağıdakı inyeksiya növlərini kimi göstərmək olar: – proseslərin inyeksiyası, kitabxanaların inyeksiyası, yüklənmədən qabaq mühitin inyeksiyası [6].

Funksiya çağırışlarının izlənməsi (Function Call Traces) – çağırılan funksiyaların nəticələrinin izlənilməsi function call trace adlanır. Bu izləmə mərhələsində çağrılan funksiyaların sırası məlumdur. Bu informasiya bizə zərərli proqramın necə işlədiyi haqqında məlumat verir.

Funksiya parametr analizi – statik analizdə funksiya parametrlərinə baxılaraq (tiplərinə və qiymətlərinə) mümkün olan parametr qiymətləri arasında hansının olma ehtimalı hesablanır. Dinamik analizdə isə funksiya çağırışı edilən giriş qiyməti əldə edilir [7].

İnformasiya axının izlənilməsi (Information Flow Tracking) - proqram tərəfindən məlumatın necə emal

edildiyinə baxılır. Sətir–sətir kod aktivləşdirmə prosesi (debug) zamanı verilənlər və ya “register” verilənləri nişanlanaraq informasiyanın hərəkəti izlənilir. Bu izləmənin məqsədi vacib verilənləri tapmaq və ya qarışıq kodun sadələşdirilməsini təmin etməkdir.

Təlimat izləmə (Instruction Trace) - Bu üsulda analiz edən şəxs sətr–sətr kod aktivləşdirmə (debug) ilə kod da olan ən alt səviyyə maşının başa düşəcəyi əmrləri analiz etməyə çalışır. Analiz vaxtı müəyyən bir iş görən maşın koduna rast gəlindikdə daha yüksək səviyyədə görmədiyi vacib məlumat bu səviyyədə əlçatan olur [8].

IV. ZİYANKAR PROQRAMLARIN VİRTUAL MÜHİTDƏ ANALİZİ

Zərərli proqramların virtual mühitdə analizi əsasən “sandbox”-larla aparılır.

SandBox – Zərərli proqramların aktivləşdirilərək, fəaliyyətlərinin avtomatik olaraq analiz edən virtual mühitdir. Sandbox mühitini qapalı bir qutu kimi təsəvvür etmək olar, bu qutunun içərisində zərərli proqramın yoluxa biləcəyi bütün şərait mövcuddur. Bu qutunun içərisindəki quruluşa baxdıqda, sandbox mühiti əsasən iki hissədən ibarət olur: analiz edilən və hesabatın hazırlandığı mühit. Əsasən ubuntu kimi linux əməliyyat sistemlərindən istifadə edilir. Analiz edilən bölümdə isə virtual mühit ilə əlaqəni təmin edən proqramlar, skriptlər və hesabatı təmin edən alətlərdən başqa əlavə əməliyyatlar aparılmır [9].

Virtual laboratoriya mühitində zərərli proqramların hərəkətlərini analiz etmək üçün lazım olan proqramlar: şəbəkə avadanlıqlarının analizi üçün – wireshark, windump, tcpdump, etheral, fayl sistemini izləmək üçün proqramlar – The Sleuth Kit, Registry (qeyd dəftəri) qeydlərini izləmək üçün proqramlar – process bat, captureBAT, regshot, xidmətləri və yaddaşın izlənilməsi üçün proqramlar – Volatility Framework-dan istifadə olunur [10].

Bundan başqa veb üzərində analiz edən Anubis, Norman, Comodo və.s kimi sandbox sistemləri verilmiş zərərli proqram nümunəsini öz verilənlər bazalarındakı imza nümunələri ilə müqayisə edib daha əvvəldən aşkar edilmiş bir zərərli proqram olub-olmadığını müəyyən edə bilirlər. Bu alətlərin yaratdığı çıxış informasiyaları sandbox kodu tərəfindən yığılaraq hesabat vəziyyətinə gətirilir.

NƏTİCƏ

Ümumiyyətlə zərərli kod sistem istifadəçinin xəbəri olmadan zərərli funksiyaları həyata keçirmək üçün hazırlanmışdır. Viruslar, soxulcanlar, troya atları, zərərli mobil kod və hücumlar daxil olmaqla bir sıra zərərli kod kateqoriyaları vardır. Zərərli proqramlar eyni zamanda tələ–funksiyalar, rutkit, klaviatura reyestrləri və cəsus kimi istifadə edilən izləmə kuki faylları kimi hücumları ehtiva edir.

Zərərli proqramların yoluxma üsulları əsasən aşağıdakı kimi olur:

–Şəbəkə üzərində olan kompüterlər axtarılır və bu kompüter üzərində olan xidmətlərin boşluqlarından istifadə edərək sistemə giriş əldə edilir.

–İnternet brauzerinin boşluğundan istifadə edərək zərərli kodun internetdən yüklənməsini təmin edir və kompüteri yoluxdurur. Hücumçuların istifadə etdiyi alətlər əsasən brauzer əlavələrində istifadə olunan jscript, vbscript kimi sistem tərəfindən aktivləşdirilə bilən skriptlərdir.

– Sosial mühəndislik, maillə də aldadılaraq istifadəçiyə fayl açmağı məcbur etmə, pulsuz proqram adı altında fayl yüklənməsi ən çox yayılan üsullardan biridir.

Kompüter sistemlərinin tətbiqi və istifadə sahələri artdıqca onlara olan təhlükələrin və zərərli proqramlarla hücumların sayı artır. Belə məsələlərə hazırlıqlı olmaq üçün kompüter şəbəkələrinin təhlükəsizliyinin qiymətləndirilməsi, zərərli proqramların analiz edilərək aşkarlanması və fəaliyyət sxeminin aydınlaşdırılması əsas məsələlərdən birinə çevrilir.

Müasir sandbox alətlərindən istifadə edərək zərərli proqramları avtomatik analiz etmək mümkündür. Bu analizlərin nəticələrinə uyğun olaraq mövcud kompüter şəbəkələrinin cari təhlükəsizlik vəziyyətini yüksəltmək mümkündür.

ƏDƏBİYYAT

[1] D. Ucci, L. Aniello, R .Baldoni, “Survey of Machine Learning Techniques for Malware Analysis”, Computers & Security, vol. 65, 2018, pp. 1627–1636

[2] M. F. Razak, N. B. Anuar, R. Salleh, A. Firdaus, “The rise of “malware”: Bibliometric analysis of malware study”, Journal of Network and Computer Applications, vol. 75, 2016, pp. 58-76

[3] J. Stiborek, T. Pevný, M. Rehák, “Probabilistic analysis of dynamic malware traces”, Computers & Security, vol. 74, 2018, pp. 221-239

[4] C. Lin, H. Pao, J. Liao, “Efficient dynamic malware analysis using virtual time control mechanics”, Computers & Security, vol. 73, 2018, pp. 359-373

[5] M. Wagner, A. Rind, N. Thür, “Wolfgang Aigner, A knowledge-assisted visual malware analysis system: Design, validation, and reflection of KAMAS”, Computers & Security, vol. 67, 2017, pp. 1-15

[6] S. Banin, G. Dyrkolbotn, “Multinomial malware classification via low-level features”, Digital Investigation, vol. 26, 2018, pp. 107-117

[7] S. P. Choudhary, D. Vidyarthi, “A Simple Method for Detection of Metamorphic Malware using Dynamic Analysis and Text Mining”, Procedia Computer Science, vol. 54, 2015, pp. 265-270

[8] M. A. Kumara, C. D. Jaidhar, “Automated multi-level malware detection system based on reconstructed semantic view of executables using machine learning techniques at VMM”, Future Generation Computer Systems, vol. 79, 2018, pp. 431-446

[9] M. Cafasso, M. Tarral, “Designing flexible sandboxing solutions to adapt to new malware trends”, Computer Fraud & Security, vol. 2018, 2018, pp. 5-9

[10] R. Islam, R. Tian, L. M. Batten, S. Versteeg, “Classification of malware based on integrated static and dynamic features”, Journal of Network and Computer Applications, vol. 36, 2013, pp. 646-656.

ANALYSIS OF MALWARE IN COMPUTER NETWORKS

Tural Yunusov
Institute of Information Technology of ANAS
Baku, Azerbaijan
turaly@mail.ru

Abstract – The article provides information about computer network security, malware, malware analysis methods, static analysis method, dynamic analysis method, virtual analysis of malicious activity and malware infection methods.

Keywords - computer networks, malware, rootkits, malicious mobile code, cookie files, hooking, sandbox.