MONITORING AND ANALYSIS OF USERS USING DLP SYSTEM

Peter László

Bachelor Degree Programme (3), FIT BUT

E-mail: xlaszl00@stud.fit.vutbr.cz

Supervised by: Michal Drozd

E-mail: idrozd@fit.vutbr.cz

Abstract: This paper presents the principles of data loss prevention systems and each element of it. It explains the term of „hooking“ in computer programming for altering the behavior of computer programs. Furthermore, it describes how the author links this technique with basic ideas of DLP and creates an endpoint-like application which purpose is to prevent unauthorized use of confidential data.

Keywords: DLP, hooking, data loss prevention

1. ÚVOD

S rozšírením počítačov v osobných a pracovných oblastiach života rástla aj kybernetická kriminalita a spolu s ňou aj potreba ochrany dát. Existuje široká paleta bezpečnostných programov: firewall, intrusion detection systémy, intrusion prevention systémy, antivírusové produkty atď., ale ani jeden z nich nie je dostatočne zameraný na ochranu dát. Preto môže dôjsť k internému úniku spôsobenou nevhodným chovaním užívateľa alebo útočníka. Práve na monitorovanie citlivých dát existujú špecializované Data Loss Prevention (ďalej len DLP) systémy.

Cieľom práce je popísať DLP systémy, navrhnúť a implementovať Endpoint aplikáciu s lokálnou centrálnou správou, ktorá bude schopná pracovať efektívne bez ostatných častí DLP.

2. DATA LOSS PREVETION SYSTÉMY

DLP sú systémy, ktoré lokalizujú, katalogizujú, monitorujú a chránia citlivé dáta podľa nastavenej politiky. Komplexnosť DLP systémov spočíva v 3 vrstvovej ochrane. Monitorujú prácu na koncových zariadeniach (Data in use), dátové toky na sieti (Data in motion) a dátové sklady (Data at rest)[3].

2.1. DATA AT REST

Základná funkcia DLP je identifikácia miesta uloženia citlivých dát. To znamená schopnosť vyhľadania vybraných súborov, ako napríklad multimediálnych súborov alebo textových dokumentov, bez ohľadu na to, či sú na serveroch alebo SANsoch (Storage Area Network). Po úspešnom nájdení systém potrebuje otvoriť a preskenovať súbor a rozhodnúť či obsah treba chrániť alebo nie. Na tento účel sú používané tzv. crawleri. Crawler je aplikácia pre skenovanie dát na koncových zariadeniach. Zbieranie týchto informácií je kľúčová pre úspešnú ochranu.

2.2. DATA IN MOTION

Na zachytávanie a analýzu sieťového toku sú využívané špecifické zariadenia. Súbory poslané cez sieť sú obecne rozdelené do paketov. DLP systém preto musí byť schopný na:

- pasívne monitorovanie;

- rozpoznanie korektného dátového toku;
- zhromaždenie nazbieraných paketov;
- rekonštrukciu súboru;
- analýzu.

Na uskutočnenie týchto krokov slúži proces DPI (deep packet inspection). Keď citlivé dáta sú detekované v toku s neautorizovaným cieľom, tak DLP je schopný zaznamenať alebo blokovať tok. Dôležitou podmienkou je dešifrovaná forma dát. V prípade nesplnenia podmienky systém DLP musí byť schopný dešifrovať dáta sám (potrebuje poznať metódu šifrovania a kľúč).

2.3. DATA IN USE

Systém v tejto časti monitoruje pohyb dát vyplývajúcich z akcií užívateľov na koncových zariadeniach, ako kopírovanie dát na USB disk, posielanie dát cez peer-to-peer aplikácie, či copy-paste. Táto časť je uskutočnená pomocou tzv. agenta. Implementovanie sád pravidiel na koncových zariadeniach má svoje limitácie. Koncový systém musí byť schopný spracovať pravidlá v prijateľnom časovom úseku.

3. HOOKING

Pri vytváraní Endpoint aplikácie je potrebná metóda pomocou ktorej je program schopný monitorovať a reagovať na udalosti efektívne a bez zaťaženia PC. Pre náš cieľ je možné použiť hooking. Hooking predstavuje techniku, ako získať kontrolu nad vykonaním určitého kódu. V operačných systémoch Windows sú implementované ako dynamicky linkované knižnice (DLL) alebo ako ovládače. V prípade použitia sa musíme rozhodnúť o spôsobe monitorovania, o aplikovania hookov a o používanom mechanizme.

3.1. TYP MONITOROVANIA

Podľa účelu aplikácie môžeme monitorovať vybraný program alebo celý systém (system-wide hooking). Pri hookovaní jedného programu sú pozorované len jeho systémové volania, zatiaľ čo system-wide hooky monitorujú vybrané udalosti bez ohľadu na volajúcu aplikáciu.

3.2. APLIKOVANIE DLL HOOKOV

V operačnom systéme Microsoft Windows každý proces má svoj privátny adresový priestor, kde sú uložené adresy virtuálnej pamäti, ktoré proces môže používať [2]. Aby vytvorená knižnica obsahujúca implementáciu hookov bola využívaná, je treba ju aplikovať do privátneho adresového priestoru. Používajú sa 3 spôsoby aplikovania [1]:

1. Upravením registrov.
2. Technika System-wide windows hooking.
3. Pomocou API funkcie LoadLibrary() a CreateRemoteThread().

3.3. APLIKOVANIE SYS HOOKOV

Kernel hooky môžeme aplikovať ako Windows service, ktoré sú spustiteľné súbory so špecifickou funkcionalitou. Môžu byť nakonfigurované tak, aby boli spustené v pozadí po štartovaní operačného systému.

3.4. MECHANIZMUS ZACHYTENIA

Podľa miesta aplikovania rozlišujeme dva typy hookov: Kernel a User level hook [4]. Rozdiel medzi implementáciou dvoch typov je, že na úrovni kernel je kód zabalený ako kernel-mode driver,

kým na úrovni user je využívaná user-mode DLL. Z pohľadu bezpečnosti a efektívnosti sú kernel level hooky výhodnejšie, avšak náročnosť implementácie je u týchto typov hookov radovo vyšší ako u user level hookov.

4. NÁVRH APLIKÁCIE

Po predchádzajúcich znalostiach sme schopní vytvoriť návrh personálneho DLP. Aplikácia bude podľa nastavenej politiky monitorovať a reagovať na vybrané systémové udalosti. Skladá sa z troch hlavných častí. Ovládač filter.sys obsahuje jadro celého programu. Jeho úlohou bude odchytávanie a filtrovanie IRP_MJ správ podľa pravidiel. Ovládač je realizovaný ako file-system mini filter. Druhá časť je filter_app.exe. Rolou užívateľskej časti aplikácie je komunikácia s ovládačom a spravovanie pomocných súborov. Spravovanie týchto súborov bude umožnené až po zadania správneho hesla do užívateľskej aplikácie. Komunikácia je dvojfázová. V prvej fáze užívateľská aplikácia pošle pravidlá ovládaču po dešifrovaní konfiguračného súboru. V druhej fáze ovládač posiela zaznamenané či zakázané udalosti, ktoré užívateľská aplikácia uloží do log súboru. S týmto sme predstavili aj pomocné súbory aplikácie (konfiguračný súbor a log súbor), ktoré z dôvodu bezpečnosti budú šifrované. Konfiguračný súbor je úložisko pravidiel, kde sú definované reakcie aplikácie na súbory a udalosti. Udalosti budú IRP_MJ správy ako napr. IRP_MJ_CREATE. Bude možnosť si vybrať z troch reakcií: zaznamenanie udalosti (silent), zaznamenanie udalosti s varovaním (warning) a zakázanie udalosti (strict). Tretia časť je skript, ktorý spĺňa úlohu crawlera (viz. kap. 2.1). Cieľom je automatické vyhľadávanie a označenie citlivých textových súborov pomocou zadaných regulárnych výrazov.

5. ZÁVER

Hlavným cieľom tohto dokumentu je oboznámiť čitateľa s DLP systémami a s technikou Hooking, pomocou ktorej je možné implementovať Endpoint časť aplikácie. Ďalej navrhnúť program, ktorý na princípe DLP bude schopný zabrániť krádeži alebo strate dát z personálnych počítačov. Aplikácia PersonalDLP je v stave vývoja. Počet možností o vylepšení a rozšírení návrhov a základnej verzie je veľká, ale primárnym cieľom je vytvoriť efektívny a použiteľný program, z ktorého vybrané časti budú prezentované na súťaži STUDENT EEICT 2011. Navrhované riešenie je priebežne konzultované s odborníkom na danú problematiku spoločnosti TrustPort a.s.

V závere je nutné sa zmieniť o tom, že ani jeden bezpečnostný program nefunguje stopercentne a ani DLP systémy nie sú výnimkou. S využitím DLP sa riziko straty citlivých dát výrazne znižuje, je však odporúčané aplikovať aj ďalšie obranné programy pre úspešnú ochranu, ako firewall či antimalwareové aplikácie.

REFERENCE

[1] IVANOV, Ivo. *Codeproject.com* [online]. 2002, 2002-12-3 [cit. 2011-03-01]. API hooking revealed. Dostupné z WWW: <http://www.codeproject.com/KB/system/hooksys.aspx>.

[2] RICHTER, Jeffrey. *Programming Applications for Microsoft Windows*. 4. Redmond, WA, USA : Microsoft Press, 1999. 1200 s. ISBN 978-1572319967.

[3] *Data Leak Prevention* [online]. 2010-09-14 [cit. 2011-03-01]. Isaca.org. Dostupné z WWW: <http://www.isaca.org/Knowledge-Center/Research/ResearchDeliverables/Pages/Data-Leak-Prevention.aspx>.

[4] RUSSINOVICH, Mark; SOLOMON, David. *Windows Internals : Including Windows Server 2008 and Windows Vista*. 5. Redmond, WA, USA : Microsoft Press, 2009. 1263 s. ISBN 0-7356-2530-1.