NETWORK TRAFFIC OBFUSCATION FOR IDS DETECTION AVOIDANCE

Daniel Ovšonka
Master Degree Programme (2), FIT BUT
E-mail: xovson00@stud.fit.vutbr.cz

Supervised by: Kamil Malinka
E-mail: malinka@fit.vutbr.cz

Abstract: This paper deals with the principles of network traffic obfuscation, in order to avoid its detection by the Intrusion Detection System installed in the network. The outcome of the work is represented by a library, that provides all the implemented techniques for further use. The library can be well utilized in penetration testing of the new systems or used by the attacker.

Keywords: Obfuscation, Intrusion Detection System, Security, Protocol masking, Network attacks

1 ÚVOD

Táto práca sa zaoberá analýzou možností obfuskácie sieťovej komunikácie, ktorej cieľom je vyhnutie sa detekcii pomocou inštalovaného IDS systému v sieti. Jedná sa o rýchlo rozvíjajúce sa odvetvie, spadajúce do počítačovej bezpečnosti. Rýchly vývoj je spôsobený hlavne tým, že dochádza k zdokonaľovaniu nasadzovaných IDS systémov, ktoré nútia útočníkov vytvárať sofistikovanejšie spôsoby obfuskácie útokov, prípadne celej sieťovej komunikácie.

Hlavným cieľom je implementácia knižnice, ponúkajúcej rôzne obfuskačné techniky, ktoré sa zameriavajú na maskovanie celej komunikácie určitým protokolom. Dôraz pri návrhu je kladený na jednoduchosť rozhrania knižnice, rozšíriteľnosť a efektívnosť jej používania. Druhým podstatným kritériom je snaha o čo najvyššiu univerzálnosť, to znamená, že knižnica by mala byť schopná úspešne obfuskovať tok voči rôznym IDS systémom s rozličným spôsobom detekcie.

2 OBFUSKAČNÉ TECHNIKY

Obfuskáciou v kontexte tejto práce rozumieme, neformálne povedané, transformáciu sieťovej komunikácie tak, aby dáta nezmenili svoju sémantiku a na druhej strane došlo k modifikácii ich syntaxe, za účelom vyhnutiu sa odhalenia bezpečnostným detekčným systémom. V súčasnosti sa používajú najmä techniky postavené na kryptografii, genetických algoritmoch, prípadne špecifické metódy zamerané priamo na maskovanie protokolov, vytvorenie polymorfného shellkódu a podobne [1].

2.1 KRYPTOGRAFICKÉ METÓDY

Kryptografické metódy [2] kladú dôraz na anonymizovanie dát. Na druhej strane nie je utajenie dát a vyhnutie sa detekcii v sieti hlavným cieľom, preto sú dáta obfuskované tak, aby nemohla byť odhalená identita jednotlivých komunikujúcich entít. Princíp spočíva v tom, že dáta (v tomto prípade najčastejšie hlavičky sieťového paketu) sú zašifrované pomocou špeciálnej funkcie, ktorá vytvára mapovanie IP adries vzťahom kardinality $N : 1$ [3] tak, aby útočník nebol schopný zrekonštruovať identity jednotlivých entít v sieti. Táto metóda avšak nie je vhodná pri obfuskácií celej sieťovej komunikácie, pretože jednoduché zašifrovanie nie je dostačujúce [4].

2.2 MASKOVANIE PROTOKOLU

Maskovanie protokolu [5] je postavené na princípe *steganografie*, keď sú dáta zo zdrojového protokolu vkladané do dát cieľového protokolu tak, aby sa komunikácia cieľového protokolu javila pre bezpečnostný systém ako štandardná a validná. Ako zdrojový protokol volíme ten, ktorý chceme zamaskovať (najčastejšie zablokovaný v danej sieti) a ako cieľový protokol vyberáme taký, ktorý je rozšírený a značne využívaný v danej sieti, aby sme mali istotu, že nedôjde aj k jeho zablokovaniu. Metóda postavená na maskovaní protokolu je v návrhu knižnice použitá ako východisková metóda, ktorá je vo vlastnej implementácii doplnená aj o princípy predošlého kryptografického prístupu.

3 NÁVRH A TESTOVANIE

Návrh knižnice kladie dôraz na efektívne a jednoduché použitie užívateľom, ktorý si ani nemusí byť vedomý vlastného fungovania a implementácie obfuskácie, keď v počiatočnej fáze postačuje, aby užívateľ inicializoval samotnú knižnicu a definoval parametre vykonávanej obfuskácie. Užívateľ zvolí napríklad, aký protokol má byť maskovaný a akým spôsobom sa obfuskácia bude vykonávať.

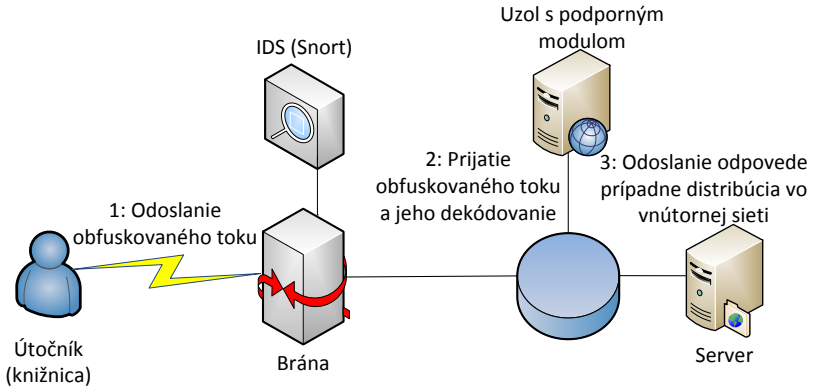
Po spustení zabezpečí knižnica presmerovanie komunikácie spadajúcej pod danú obfuskáciu priamo z jadra operačného systému, cez virtuálne fronty paketov, na výstup a všetky odpovedajúce pakety hostiteľského stroja budú kontrolované knižnicou. Pakety jednotlivých front (vstupná, výstupná) sa analyzujú samostatnými vláknami a ak paket podlieha pravidlám obfuskácie, dôjde k jeho transformácii, vypočítajú sa nové kontrolné súčty. Ak je potrebné, paket môže byť pozdržaný, aby sa modifikovali aj časové odozvy medzi odosielaniami a konečne sa paket odošle na výstup (respektíve prichádzajúci paket sa prepošle do systému). V prípade paketu, ktorý nepodlieha obfuskácii, sa jednoducho nezmenený prepošle ďalej, aby sme zabezpečili transparentnosť voči vyšším vrstvám a operačnému systému. Finálna fáza spočíva v ukončení činnosti, keď dôjde k odobratiu presmerovania sieťovej komunikácie a navrátenie obsluhy jadru systému, čím sa vráti systém do pôvodného stavu.

3.1 TESTOVACÍ SYSTÉM

Na účely testovania bolo vytvorené prostredie z virtuálnych počítačov, rozdelené na dve samostatné podsiete, ktoré sú prepojené sieťovou bránou, chránenou systémom IDS. Podsiete simulujú vnútornú chránenú sieť a okolitý svet, pričom všetka komunikácia smerovaná medzi nimi je kontrolovaná pomocou IDS na bráne. Ako referenčný model IDS bola zvolená implementácia *Snort* (http://www.snort.org), ktorá je v súčasnosti, ako jedna z mála, poskytovaná ako *open-source* a patrí k najrozšírenejším systémom, ktorý je často základom mnohých komplexnejších riešení. Základná detekcia je postavená na odchytávaní paketov v promiskuitnom režime a porovnávaní so *signatúrami*, ktorá je doplnená o kontrolu protokolov a sieťových anomálií. Z tohto dôvodu nie je pri vyhýbaní sa detekcii vhodné použiť jednoduché *šifrovanie* (2.1), pretože vyhodnocovanie prebieha aj napríklad na základe veľkostí paketov alebo časov medzi ich doručením. Útočník, využívajúci knižnicu, primárne komunikuje s jedným hostiteľským počítačom vo vnútri siete, na ktorom sa nachádza podporný modul, ktorý rieši obnovenie obfuskovaného toku na pôvodné dáta. Distribúciu tohto podporného modulu táto práca primárne nerieši a predpokladá sa, že modul je už v sieti zavedený. Schému môžeme vidieť na obrázku 1.

3.2 METODIKA TESTOVANIA A VÝSLEDKY

Testovanie spočíva v dvojitom zaslaní zhodnej postupnosti paketov, obsahujúcej kompromitujúce dáta, na ktoré by mal IDS zareagovať. Ako prvá je zaslaná originálna postupnosť, ako druhá postupnosť, ktorá bola modifikovaná obfuskačnou knižnicou. Úspešnosť posudzujeme na základe počtu alarmov, ktorým sa na výstupe IDS podarilo touto transformáciou zamedziť. V najlepšom prípade nedôjde pri obfuskovanom toku ku generovaniu žiadnych hlásení.



Obrázek 1: Architektúra virtuálnej testovacej siete.

Výsledky môžeme demonštrovať na komunikácii pomocou protokolu `telnet`, ktorá je v prípade prístupu z vonkajšej siete monitorovaná a Snort generuje alarmy, napríklad pri chybnom pokuse o prihlásenie. Náročnejšiu obfuskáciu predstavuje ukrytie *skenovania otvorených portov* na počítači vo vnútornej sieti, ktorá je vykonávaná pomocou programu `nmap`. Snort, v tomto prípade s použitím aktuálnych pravidiel, vygeneruje sedem alarmov, keď je schopný odhaliť, že sa jedná o *TCP Portscan*. Okrem toho zachytí aj podozrivú komunikáciu na niektorých portoch. V oboch týchto prípadoch, pri použití obfuskačnej knižnice, dôjde k transformácii odchádzajúcej komunikácie tak, že IDS nezachytí žiadne podozrivé pakety, nevygeneruje žiadne alarmy a ani neuloží informácie do logovacieho súboru na prípadnú kontrolu.

4 ZÁVER

Čitateľ bol v tejto práci oboznámený s princípmi obfuskácie sieťovej komunikácie, ktoré slúžili ako odrazový mostík pri návrhu knižnice. Výsledná knižnica môže byť ďalej používaná pri penetračnom testovaní nasadzovaných systémov, prípadne pri skúmaní schopností detekcie IDS systému. Možné využitie by knižnica mohla nájsť aj pri anonymizovaní a skrývaní sieťovej komunikácie protokolom, ktorý je v sieti nežiadúci alebo blokovaný. Testovanie ukazuje pomerne vysokú úspešnosť obfuskácie základných protokolov voči IDS, nevýhodou však je, že testy prebiehajú iba voči implementácii Snort, pretože pri iných riešeniach sa jedná o komerčné, voľne nedostupné produkty.

REFERENCE

[1] Kayacik, H., Zincir-Heywood, A., Heywood, M. Automatically Evading IDS Using GP Authored Attacks. In Computational Intelligence in Security and Defense Applications, 2007, CISDA 2007.

[2] Hessler, A., Kakumaru, T., Perrey, H. et al. Data Obfuscation with Network Coding. Computer Communications, 2010, vol. 35, no. 1, s. 48-61, ISSN 0140-3664.

[3] Riboni, D., Villani, A., Vitali, D. et al. Obfuscation of Sensitive Data in Network Flows. In INFOCOM, 2012 Proceedings IEEE, 2012, s. 2372-2380, ISSN 0743-166X.

[4] Hjelmvik, E., John, W. Breaking and Improving Protocol Obfuscation. Sweden: Chalmers University of Technology and Göteborg University, 2010, Technical report.

[5] Mohajeri Moghaddam, H., Li, B., Derakhshani, M. et al. SkypeMorph: Protocol Obfuscation for Tor Bridges. In Proceedings of the 2012 ACM conference on Computer and Communications Security, New York, NY, 2012, s. 97-108.