elektrorevue
ISSN 1213 - 1539

Rok / Year:	Svazek / Volume:	Číslo / Number:
2010	12	6

Základní techniky skenování TCP portů

Basic techniques of TCP ports scanning

Ondřej Rášo, Jiří Sobotka

ondra.raso@phd.feec.vutbr.cz

Fakulta elektrotechniky a komunikačních technologií VUT v Brně

Abstrakt: Článek pojednává o základních technikách skenování portů protokolu TCP (Transmission Control Protocol) a možnostech obrany proti nim. Nejprve je v článku krátce popsán protokol TCP. Jsou popsány důležité informace v hlavičkách protokolů IP (Internet Protocol) a TCP a je také popsán postup při založení TCP spojení. Hlavní těžiště článku je v popisu základních technik skenování TCP portů. Konkrétně se jedná o techniky tajného skenování, skenování FIN, X-mas a NULL a v neposlední řadě techniky tzv. nečinného skenování. Veškerý popis těchto technik je doplněný o výpisy zachycených paketů síťovým analyzátorem. V závěrů článku jsou navrhnuty metody aktivní obrany proti popsaným typům skenování.

Abstract: In the article, the basic techniques of TCP ports scanning are describe. Firstly, the most important information stored in IP protocol headers are mentioned and then the established of TCP connection is described. The article main is placed in the description and testing of the TCP ports scanning (FIN scan, X-mas scan, NULL scan and idle scaning).

elektrorevue
ISSN 1213 - 1539

Základní techniky skenování TCP portů

Ondřej Rášo, Jiří Sobotka

Fakulta elektrotechniky a komunikačních technologií VUT v Brně
Email: ondra.raso@phd.feec.vutbr.cz

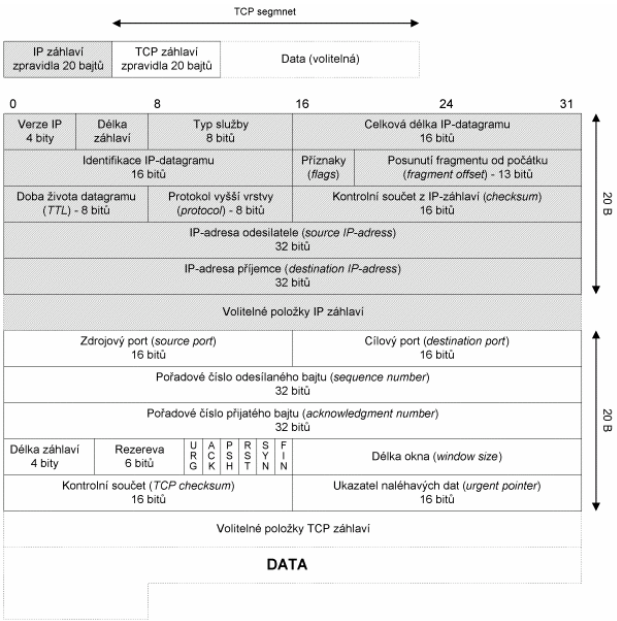
Abstrakt *– Článek pojednává o základních technikách skenování portů protokolu TCP (Transmission Control Protocol) a možnostech obrany proti nim. Nejprve je v článku krátce popsán protokol TCP. Jsou popsány důležité informace v hlavičkách protokolů IP (Internet Protocol) a TCP a je také popsán postup při založení TCP spojení. Hlavní těžiště článku je v popisu základních technik skenování TCP portů. Konkrétně se jedná o techniky tajného skenování, skenování FIN, X-mas a NULL a v neposlední řadě techniky tzv. nečinného skenování. Veškerý popis těchto technik je doplněný o výpisy zachycených paketů síťovým analyzátorem. V závěrů článku jsou navrhnuty metody aktivní obrany proti popsaným typům skenování.*

1 Úvod

Skenování TCP portů, dále jen skenování, je proces, kdy jsou sledovány aktivní služby na síťovém rozhraní. Služby v kontextu toho článku budou nazývány síťové programy které mohou operovat se svým síťovým rozhraním popsaným IP adresou. Dále jsou tyto služby adresovatelné svým jedinečným číslem TCP portu, jedinečným v rámci jednoho síťového rozhraní. Aktivními službami mohou být například ftp nebo webové servery, služby vzdálené plochy ve Windows popř. i trojské koně nebo jiné škodlivé programy nainstalované na počítači.
Jelikož skenování může být považováno za jistých situací za nelegální činnost, je potřeba zmínit právní aspekty tohoto jednání [1]. Důvodů skenování může být hned několik [2]. Prvním důvod pro skenování může být čistě uživatelská snaha zjistit jaké služby jsou aktivní. Skenování totiž může odhalit i systému skryté služby, viz různé typy škodlivých programů. Dalším důvodem je administrátorská snaha zjistit všechny otevřené porty ve všech administrátorem spravovaných stanicích. Díky skenovaní lze například vhodným skriptem několikrát denně zjišťovat aktivní služby v síti a tento údaj pak porovnávat se staršími údaji a následně vyhodnocovat [2]. Posledním typem důvodu pro skenování je snaha útočníka zmapovat aktivní služby na počítači oběti a zjistit potenciálně slabá místa jakou jsou například starší neaktualizované verze programů atd. Jako obrana proti skenování se používají prvky aktivní obrany, některé z těchto technik jsou popsány dále v článku.
Článek je zaměřen jen na skenování portů TCP. Skenování portů UDP (User Datagram Protocol) je taky sice možné, ale použití je minimální [3]. UDP protokol totiž nesestavuje spojení, jedná se tedy o datagramovou službu velmi podobnou protokolu IP. Tento protokol je použit spíše pro přenos dat v reálném čase jako je přenos hlasu, streamovaného videa atd. Tyto aplikace většinou nepředstavují bezpečnostní rizika

2 Protokol TCP

Protokol TCP je spojovanou službou, tj. službou která navazuje spojení mezi klientem a serverem. Klient inicializuje toto spojeni a server obsluhuje požadavky klienta. Vytvořené spojení je plně duplexní, přenášené data jsou číslována a integrita je zabezpečena kontrolním součtem [4] [5]. Protokol TCP pracuje na transportní vrstvě v modelu TCP/IP a také v modelu ISO/OSI.
Před samotným výkladem o technikách skenování portů je potřeba popsat důležité informace v hlavičkách protokolů TCP a také IP, obr. 1.



Obr. 1: Hlavička protokolů TCP a IP [4].

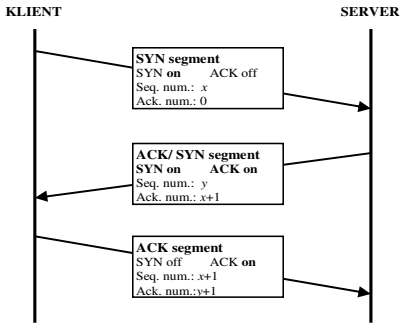
Šedou barvou je označená hlavička protokolu IP a světlou hlavička protokolu TCP. V levé horní části je zobrazen princip obalování dat, kdy k segmentu TCP, hlavičce TCP a jeho datům, je přidána hlavička protokolu IP. Velikost hlavičky protokolu IP je zpravidla 20 bajtů. Tato hlavička hlavně obsahuje IP adresu příjemce a IP adresu odesilatele IP paketu. IP paket bude dále v textu označován jako paket i když se jedná o datagram a to z důvodů užité terminologie v oblasti modelu síťové komunikace podle TCP/IP.

Světlým pozadím na obrázku 1 je označena hlavička protokolu TCP. Prvních 32 bitů je určeno pro čísla zdrojového a cílového portu. Následují pořadová čísla bajtů odesilatele a příjemce dat, anglicky jsou jejich názvy "sequence number" a "acknowledge number", podle těchto anglických názvu budou pojmenovány jejich zkratky, tedy Seq. a Ack.. Tato pořadová čísla slouží k zajištění pořadí přijímaných TCP segmentů. Z ostatních důležitých dat je potřeba zmínit příznaky TCP segmentu. Tyto příznaky slouží k vedení relace mezi klientem a serverem. V tabulce 1 jsou tyto příznaky lépe popsány.

TCP příznaky		Význam
URG	urgent	Poslední oktet urgentních dat
ACK	acknowledgment	Potvrzení spojení
PSH	push	Předání dat vyšší vrstvě.
RST	reset	Odmítnutí spojení
SYN	synchronize	Synchronizace sekvenčních čísel.
FIN	finish	Ukončení spojení

Tabulka 1: Příznaky TCP segmentu.

Na obrázku 2 je graficky zobrazen proces navázání spojení. Na začátku klient nejprve pošle tzv. SYN segment serveru. Tento segment má nastaven jediný příznak a to příznak SYN, tj. SYN má hodnotu 1. Dále obsahuje své sekvenční číslo x (dále jen Seq.) a číslo potvrzovacího TCP segmentu (dále jen Ack.) má hodnotu 0. Server odpoví ACK/SYN segmentem, segment bude mít Seq. na hodnotě y a Ack. na hodnotě x + 1. Posledním krokem je vyslání ACK segmentu. Tento segment má nastaven příznak ACK, a hodnoty Seq. a Ack. jsou x + 1 resp. y +1. Tento proces navazování spojení bývá nazýván jako TCP handshake.



Obr. 2: Navázání spojení protokolem TCP

3 Techniky skenování TCP portů

3.1 Otevřené skenování (connect scanning)

Prvním typem skenováním je otevřené skenování. Princip je velmi jednoduchý a v některých publikacích tato technika dokonce není považována za skenování [6]. Podstata spočívá v tom, že klient se úplně připojí na službu na cíleném portu. TCP porty s rozsahem 0 – 1023 jsou dobře známé porty, tedy porty které jsou určeny pro často používané služby, např. ftp server. V [7] jsou definovány všechny tyto tzv. dobře známe porty. Útočník tedy ví jaká služba na těchto portech běží a proto se může zkusit připojit. Příkladem může být pokus o připojení na ftp server, obrázky 3 a 4. Na IP adrese 192.168.0.82 neběží ftp server a pokus o připojení je zobrazen na obrázku 3. Na tomto obrázku je zobrazen logovací výpis ze sítového analyzátoru WireShark [10]. V 1. sloupci je zobrazeno číslovaní řádků, ve druhém čas zachycení dat v sekundách,

1	**192.168.0.76**	**192.168.0.82**	TCP	inova-ip-disco > ftp [SYN] Seq=0 Win=65535 Len=0 MSS=1460
2	**192.168.0.82**	**192.168.0.76**	TCP	ftp > inova-ip-disco [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
3	**192.168.0.76**	**192.168.0.82**	TCP	inova-ip-disco > ftp [SYN] Seq=0 Win=65535 Len=0 MSS=1460
4	**192.168.0.82**	**192.168.0.76**	TCP	ftp > inova-ip-disco [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
5	**192.168.0.76**	**192.168.0.82**	TCP	inova-ip-disco > ftp [SYN] Seq=0 Win=65535 Len=0 MSS=1460
6	**192.168.0.82**	**192.168.0.76**	TCP	ftp > inova-ip-disco [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Obr. 3: Pokus o připojení na ftp server pro případ neaktivního ftp serveru.

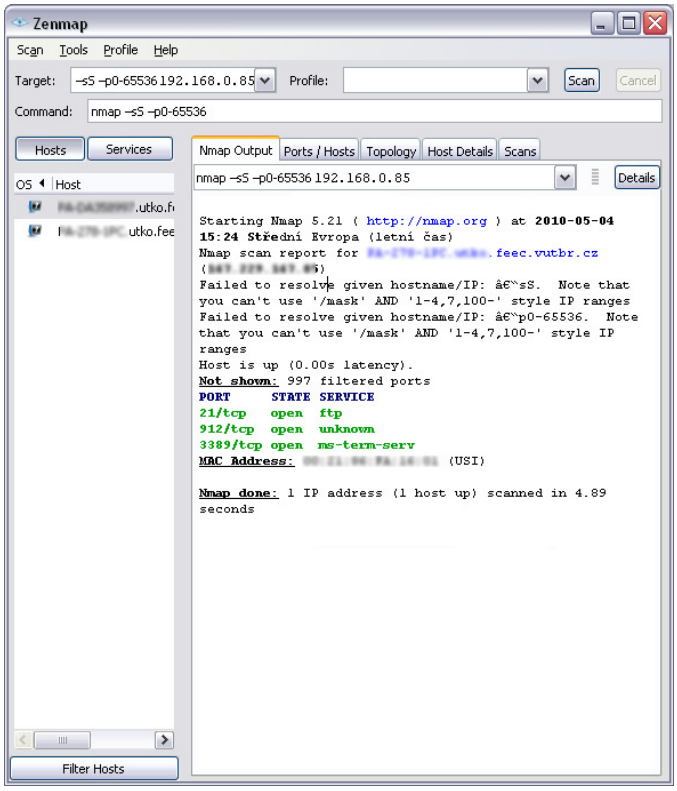
1	**192.168.0.76**	**192.168.0.85**	TCP	psbserver > ftp [SYN] Seq=0 Win=65535 Len=0 MSS=1460
2	**192.168.0.85**	**192.168.0.76**	TCP	ftp > psbserver [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460
3	**192.168.0.76**	**192.168.0.85**	TCP	psbserver > ftp [ACK] Seq=1 Ack=1 Win=65535 Len=0
4	**192.168.0.85**	**192.168.0.76**	FTP	Response: 220-Cerberus FTP Server - Personal Edition
5	**192.168.0.76**	**192.168.0.85**	FTP	Request: USER anonymous
6	**192.168.0.85**	**192.168.0.76**	FTP	Response: 331 User anonymous, password please
7	**192.168.0.76**	**192.168.0.85**	FTP	Request: PASS kraaa@kraaa.ka
8	**192.168.0.85**	**192.168.0.76**	TCP	ftp > psbserver [ACK] Seq=166 Ack=38 Win=64203 Len=0
9	**192.168.0.85**	**192.168.0.76**	FTP	Response: 530 Not logged in. Username/password incorrect, user disabled, or user logged in too many times
10	**192.168.0.76**	**192.168.0.85**	FTP	Request: QUIT
11	**192.168.0.85**	**192.168.0.76**	FTP	Response: 221 Goodbye
12	**192.168.0.85**	**192.168.0.76**	TCP	ftp > psbserver [FIN, ACK] Seq=276 Ack=44 Win=64197 Len=0
13	**192.168.0.76**	**192.168.0.85**	TCP	psbserver > ftp [ACK] Seq=44 Ack=277 Win=65260 Len=0
14	**192.168.0.76**	**192.168.0.85**	TCP	psbserver > ftp [RST, ACK] Seq=44 Ack=277 Win=0 Len=0

Obr. 4: Pokus o připojení na ftp server pro případ aktivního ftp serveru.

v 3. sloupci je zobrazena IP adresa odesilatele, ve 4. sloupci IP adresa příjemce, v 5. sloupci typ protokolu a v posledním sloupci je zobrazen krátký slovní popis zachycených dat.
Na obrázku 4 je zobrazen pokus o připojení na ftp server na IP adrese 192.168.0.85. I když se útočníkovi nepodařilo přihlásit se na tento server, tak lze z obrázku vyčíst, že služba ftp serveru je aktivní. Podobných způsobem jde vyzkoušet aktivita téměř všech portů z rozsahu 0 – 1023.
Pokud služba není aktivní (ftp server neběží) tak server odpoví RST segmentem, obr. 3 řádky 2, 4 a 6. Pokud je služba aktivní (ftp server běží) tak server odpoví ACK/SYN segmentem, obr. 4 řádek 2. V následujících řádcích z obr. 4 je zachycen síťový provoz od poslání segmentu ACK segmentu až po odmítnutí připojení na ftp server (uživatel neznal heslo a se pokusil připojit jako anonym).
Pokud uživateli není znám typ služby která je na skenovaném portu, tak je možno realizovat pouze TCP handshake, obrázek 2. Pokud žádna služba na skenovaném portu není tak server zašle RST segment, obrázek 3. Pokud ovšem je nějaká služba aktivní, tak server zašle po navázaní TCP spojení tzv. uvítací banner. Uvítací banner je uvítací hlášení, tento banner zpravidla obsahuje jméno programu a dodatečné informace např. o verzi programu, platformě nebo operačním systému. Na obrázku 4 je banner zaslán ftp server ihned po zavedení TCP spojení, řádek 4. Útočník je tedy schopen zjistit podle odpovědí serveru zda je nebo není port aktivní.
Tato metoda má jednu velkou nevýhodu. Pokud totiž proběhne navázáni TCP spojení, tak většina aplikaci si zaznamená IP adresu klienta, který vytvořil toto spojení. A to je pro případné útočníky samozřejmě nežádoucí.

3.2 Tajné skenování (stealth scanning)

Tajné skenování někdy také nazvané SYN skenování, nebo také napůl otevřené spojení [6]. Standardně útočník zašle SYN segment, server odešle SYN/ACK segment a útočník zašle RST segment. Tedy místo potvrzovacího ACK segment zašle RST segment pro ukončení spojení. Výhoda spočívá především v tom, že spojení není plně navázáno a tedy ani nedojde k uložení IP adresy útočníka do logovacích souborů, jak tomu bylo v případě otevřeného skenování. Na obrázku 5 je zobrazeno grafická nástavba programu nmap [8]. Je zde uveden výpis po tajném skenování počítače na IP adrese 192.168.0.85.



Obr. 5: Program nmap, příklad tajného skenování.

Výpisy ze sítového analyzátoru WireShark jsou zobrazeny na obrázcích 6 a 7. Na obr. 6 je zobrazen výpis síťové komunikace pro případ tajného skenování na zavřeném portu č. 21 a na obr. 7 je zobrazen výpis síťové komunikace pro případ tajného skenování na otevřeném portu č. 21. Program nmap oproti standardnímu tajnému skenování neposílá RST segmenty, ale pošle jen jeden segment a to SYN segment, obrázek 7. Server pak 3-krát zašle SYN/ACK segment a poté zašle RST segment čímž odmítne spojení.

1	192.168.0.76	192.168.0.82	TCP	41129 > ftp [SYN] Seq=0 Win=4096 Len=0 MSS=1460
2	192.168.0.82	192.168.0.76	TCP	ftp > 41129 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Obr. 6: Výpis síťové komunikace, tajné skenování na zavřeném portu č. 21

1	192.168.0.76	192.168.0.85	TCP	46279 > ftp [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2	192.168.0.85	192.168.0.76	TCP	ftp > 46279 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460
3	192.168.0.85	192.168.0.76	TCP	ftp > 46279 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460
4	192.168.0.85	192.168.0.76	TCP	ftp > 46279 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460
5	192.168.0.85	192.168.0.76	TCP	ftp > 46279 [RST] Seq=1 Win=0 Len=0

Obr. 7: Výpis síťové komunikace, tajné skenování na otevřeném portu č. 21.

1	192.168.0.76	192.168.0.82	TCP	45877 > ftp [<None>] Seq=1 Win=3072 Len=0
2	192.168.0.82	192.168.0.76	TCP	ftp > 45877 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
3	192.168.0.76	192.168.0.82	TCP	45878 > ftp [<None>] Seq=1 Win=4096 Len=0
4	192.168.0.82	192.168.0.76	TCP	ftp > 45878 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

Obr. 8: Výpis síťové komunikace, NULL skenování na zavřeném portu č. 21.

1	192.168.0.76	192.168.0.85	TCP	60104 > ftp [<None>] Seq=1 Win=3072 Len=0
2	192.168.0.76	192.168.0.85	TCP	60105 > ftp [<None>] Seq=1 Win=3072 Len=0

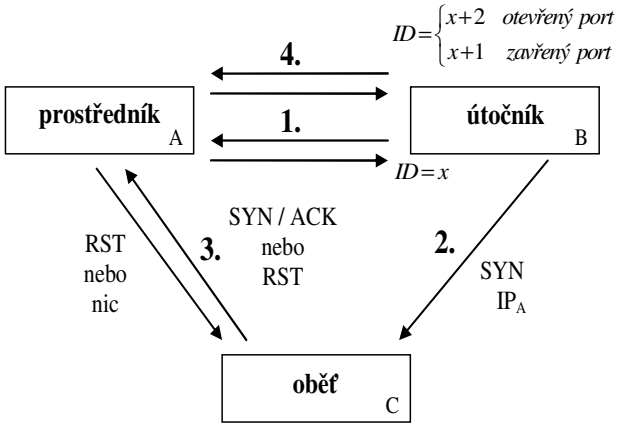
Obr. 9: Výpis síťové komunikace, NULL skenování na otevřeném portu č. 21.

3.3 Skenování FIN, X-mas a Null

Při tomto typu skenování útočník zasílá "nesmyslné" segmenty. Jedná se o segmenty s chybně nastavenými příznaky. U FIN skenování je nastaven FIN příznak, u X-mass jsou nastaveny příznaky FIN,URG a PUSH a u posledního typu NULL skenování nejsou nastaveny žádné příznaky. Pokud server dodržuje specifikaci RFC 793 [9] a současně skenovaný port není otevřený tak odpoví RST segmentem, pokud je port otevřený tak neodpoví vůbec, ignoruje přijatý segment.
Výpis síťové komunikace při NULL skenování je zobrazen na obrázcích 8 a 9.

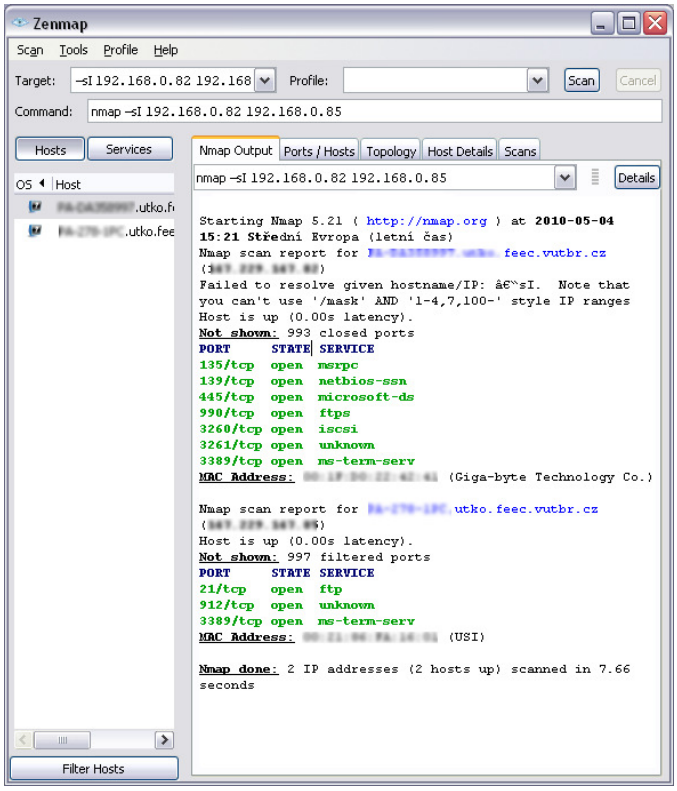
3.4 Nečinné skenování (iddle scanning)

Tento skenování využívá 16 bitové identifikační číslo (dále jen ID) IP paketů, viz obrázek 1 pole identifikace IP datagramu. Využívá se především toho, jakým způsobem dochází k inkrementaci ID. Tedy s každým odeslaným IP paketem se toto číslo změní, velikost změny (velikost inkrementace) je závislá na typu operačního systému. Například v operačních systémech typu Windows je velikost ID vždy zvětšena o 1 [3]. Blokové schéma tohoto typu skenování je zobrazeno na obrázku 10. Na tomto schématu jsou naznačené tři bloky. Útočník reprezentuje síťové rozhraní ze kterého se provádí nečinné skenování, oběť je sítové rozhraní které bude skenováno a prostředníkem je nejčastěji neaktivní počítač, je důležité aby po dobu skenování nebyl na tomto síťovém rozhraním žádný provoz. Skenování probíhá tak, že v 1. kroku útočník zjistí velikost ID a způsob inkrementace prostředníka.To udělá třeba tak, že zašle IP paket prostředníkovi. Útočníka nezajímá celá odpověď, ale jen hodnota ID. Poté zašle v kroku č.2 SYN segment s IP adresou prostředníka oběti. V 3. kroku oběť odpoví prostředníkovi. Tato odpověď bude závislá na tom, zda je skenovaný port otevřeny nebo ne. Pokud je otevřený tak oběť zašle SYN/ACK segment a prostředník na to odpoví RST segmentem, ID prostředníka se zvýší o 1. Pokud je ale skenovaný port zavřeny tak oběť zašle RST segment prostředníkovi a ten už nijak na tuto komunikaci nereaguje, ID zůstane stejné. V posledním 4. kroku zašle útočník SYN/ACK segment prostředníkovi a ten odpoví RST segmentem. Útočník v tomto kroku sleduje změnu ID. Pokud se ID změnilo o 2, tak skenovaný port oběti je otevřený, pokud se ID změnilo jen o 1 tak tento port není otevřený.



Obr. 10: Blokové schéma nečinného skenování

Na obrázku 11 je zobrazen výpis z programu nmap po nečinném skenováním, IP adresa prostředníka je 192.168.0.82 a IP adresa oběti je 192.168.0.85.



Obr. 11: Výpis z programu nmap po nečinném skenováním

4 Aktivní obrana

4.1 Obrana proti skenování FIN, X-mas a NULL

Obrana proti tomuto typu skenování je relativně jednoduchá. Stačí aby server neodpovídal RST segmentem na útočníkovi FIN, X-mas, NULL segmenty na zavřených portech. Útočník pak nedokáže rozlišit, zda je port otevřeny nebo ne. Velmi jednoduchá implementace této obrany na operačním systému LINUX drobné úpravě kernelu. Jedná se o nedodržení specifikace. Jeden z postupů je popsán např. v [6].

4.2 Obrana proti tajnému skenování

Obrana proti tomuto typu skenováním je trochu složitější. Obrana může být řešena filtrováním provozu na transportní vrstvě. Tuto filtraci nemusí nutně provádět firewall, je možno použit i skript který bude obsluhovat síťový provoz který je adresovaný na uzavřené porty. Tato obsluha by mohla fungovat tak, že pokud přijde od útočníka SYN segment na zavřený port, tak skript automaticky odpoví SYN/ACK segmentem. Port se pak bude jevit pro útočníka jako otevřený [6].

4.3 Obrana proti otevřenému skenování

Obrana může vycházet z modifikace obrany proti tajnému skenování. Skript ovšem kromě odpovědi SYN/ACK segmentem současně pošle banner virtuální aplikace. Banner by měl být takový, aby útočníka dostatečně zmátl a on tento podvrženy banner považoval za pravý [6].

4.4 Testování bezpečnosti portů

Velice často bývají určité nevyužité porty volně otevřeny, bez kontroly operačním systémem, prohlížeči nebo jinými aplikacemi. Počítač je tak zanechán ve zranitelném stavu. Většina softwarových vývojářů pravidelně vydává opravy, service packy a updaty v případě že objeví otevřený port, vytvářející bezpečnostní problém. Opravy softwarových aplikací zavírají tyto otevřené dveře do systému, proto je pravidelný update velice důležitý. Pro ověření bezpečnosti systému existují programy pro skenování portů, které je vhodné používat pro pravidelné kontroly otevřených portů počítače. K dispozici je i řada online skenerů, které prověří počítač skrze internetové připojení. Jedním z kvalitních skenerů je např. Symantec Security Check, nabízející kompletní analýzu vašeho systému.
Existuje samozřejmě i řada klasických nástrojů, využitelných k ověření celé podnikové sítě a nalezení bezpečnostních slabin. Jedním z nich je i Nsauditor [12], nabízející nástroj pro komplexní skenování sítě i portů. Pracuje s jednotlivými IP pakety, pomocí kterých dokáže zjistit dostupné počítače, jaké porty nabízejí, který operační systém používají a další důležité charakteristiky. Široká škála nabízených typů skenování umožňuje provést cílený útok na konkrétní službu či klienta, nebo provést rozsáhlé skenování celé sítě. Aplikace obsahuje širokou databázi tzv. softwarových otisků prstů, jež porovnává s otisky získanými ze otevřených portů a dokáže určit používaný operační systém či aplikaci.

4.5 Honeypot

V počítačové terminologii je výrazem honeypot [11] (hrnec s medem) označována past nachystaná na případného útočníka. V našem případě se jedná o nastrčený port, jevící se jako otevřený. Zatímco útočník ztrácí čas skenováním tohoto portu, a zjišťováním dostupných služeb, veškerá jeho aktivita je monitorována. Na rozdíl od jiných obran proti skenování portů, honeypot neřeší daný problém, pouze poskytuje čas pro standartní řešení. Port, na kterém je nasazen systém honeypot nemá žádnou autorizovanou funkci, nemá žádné využití pro uživatele. Teoreticky by na tomto portu neměl být žádný provoz. To znamená, že jakákoliv komunikace s honeypot portem je s vysokou pravděpodobností neautorizovaná nebo i škodlivá aktivita. Jakékoliv spojení s nastrčeným portem je pravděpodobně skenování, útok nebo kompromitace portu. Aplikace honeypotu má několik výhod: zachytává pouze malé množství dat, pouze ta zaslaná útočníkem. Pro jejich zpracování není třeba vysokého výpočetního výkonu. Honeypot umožňuje nastavení úrovně interakce s útočníkem. Mohou pouze emulovat služby a operační systémy, aktivita útočníka je tak limitována úrovní emulace honeypotu. Například emulovaná FTP služba naslouchající na portu 21 může pouze emulovat FTP login nebo může podporovat řadu dalších FTP příkazů. Jiný type honeypotu naopak nabízí útočníků reálný operační systém se skutečnými službami. Aplikace tohoto typu je sice náročnější, ale umožňuje získání komplexních dat o uživateli, jeho chování a způsobů útoků. Jako příklad sofistikovaného honeypotu lze uvést Symantec Decoy Server.

5 Závěr

Článek pojednával o základních technikách skenování TCP portů a možnostech obrany proti nim. Nejprve byl v článku krátce popsán protokol TCP. Dále byly popsány důležité informace v hlavičkách protokolů IP a TCP a byl popsán postup při založení TCP spojení. Ve stěžejní části článku byly popsány základní techniky skenování TCP portů. Konkrétně se jedná o techniky tajného skenování, skenování FIN, X-mas a NULL a v neposlední řadě techniky tzv. nečinného skenování. Veškerý popis těchto technik byl doplněný o výpisy zachycených paketů síťovým analyzátorem. V závěrů článku byly navrhnuty metody aktivní obrany proti popsaným typům skenování.

Literatura

[1] MATEJKA J.: *(Ne)legální port scanning*, 2001, dostupné z www: http://www.lupa.cz/clanky/nelegalni-port-scanning/

[2] HALLER M.: *Skenování portů: jaká služba nám to běží?*, 2007, dostupné z www: http://www.lupa.cz/clanky/skenovani-portu-jaka-sluzba-nam-to-bezi/

[3] HALLER M.: *Skenování portů: techniky*, 2006, dostupné z www: http://www.lupa.cz/clanky/skenovani-portu-techniky/

[4] DOSTÁLEK L.: *Velký průvodce protokoly TCP/IP: bezpečnost*, 2002, dostupné z www: http://www.cpress.cz/knihy/tcp-ip-bezp/CD-0x/9.html

[5] DOSTÁLEK L.: *Velký průvodce protokoly TCP/IP a systémem DNS*, 2002, dostupné z www: http://www.cpress.cz/knihy/tcp-ip-bezp/CD-0x/9.html

[6] ERICKSON J.: *HACKING, umění exploitace*, 263 s., Zoner Press, Brno, 2003, ISBN: 80-86815-21-8

[7] IANA, *Assigned Numbers,* dostupné z www: http://www.iana.org/assignments/port-numbers

[8] NMAP, *Nmap Reference Guide*, dostupné z www: http://nmap.org/book/man.html

[9] Defense Advanced Research Projects Agency, *Specifikace: RFC793, Transmission Control Protocol,* dostupné z www: http://www.faqs.org/rfcs/rfc793.html

[10] WIRESHARK, síťový analyzátor, dostupný z www: http://www.wireshark.org/

[11] SPITZNER L.: *Honeypots*, 2003, dostupné z www: http://www.tracking-hackers.com/papers/honeypots.html

[12] NSAUDITOR, síťový skener, dostupný z www: http://www.nsauditor.com