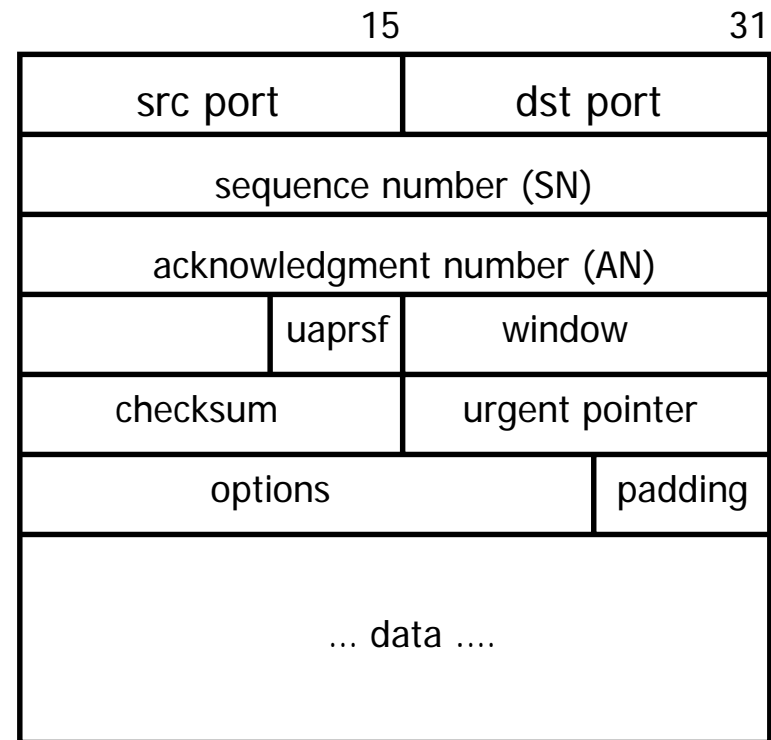
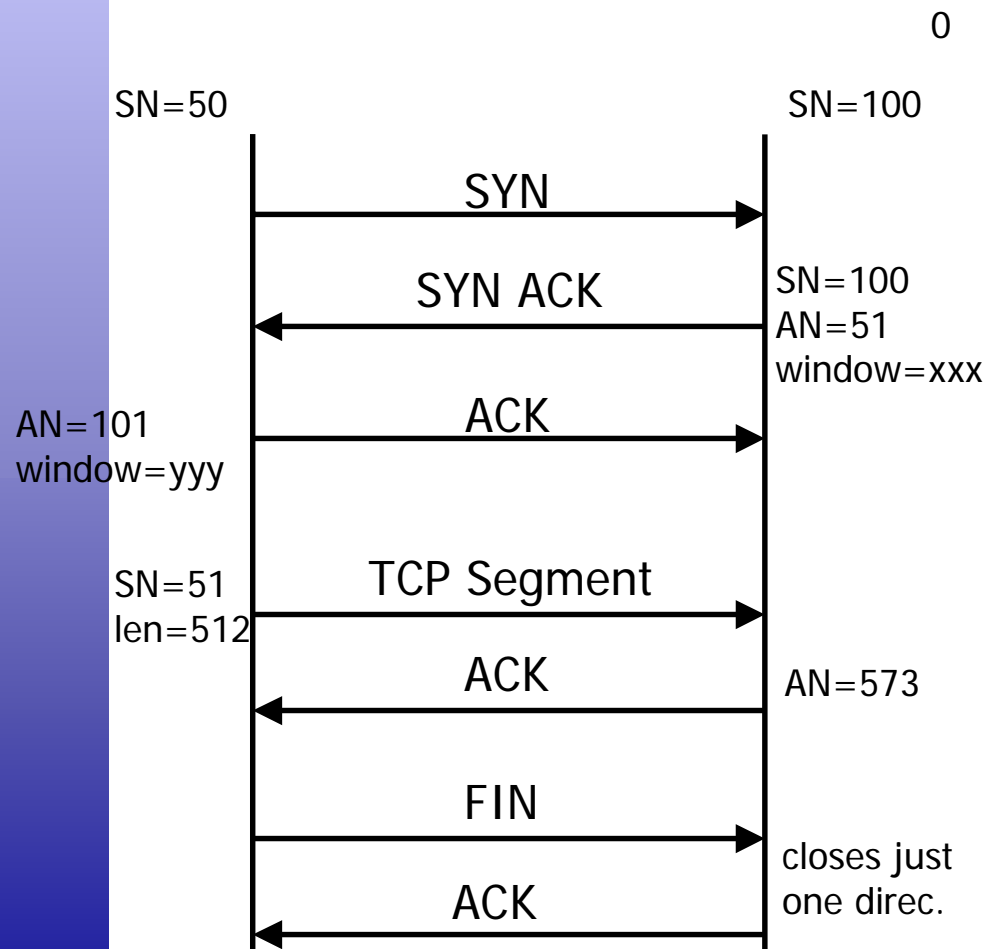




TCP – resetovací útok

Richard Latislav, **Dan Cvrček**
Vysoké učení technické v Brně

TCP protokol - základy



u – urgent
a – acknowledge
p – push

r – reset
s – syn (chronise)
f – fin

Parametry spojení

- spojení je identifikováno na úrovni TCP portů
- acknowledge a sequence no. číslují bajty, nikoliv pakety
- window – podle protokolu, dynamicky se mění podle množství přenášených dat
 - např. ssh 2000-6000, ftp, scp při přenášení dat 65535
- window 64 kB je příliš malé pro dnešní vysokorychlostní sítě
 - zvětšování oken pomocí multiplikátorů
 - window scaling (0-14 bits) => max size 1GB

Slabá místa TCP

- identifikace klienta je na základě IP adresy
 - tu lze libovolně změnit, nebo vytvořit vlastní paket s libovolným obsahem
- pro vytvoření packetu v Linux, Unix je třeba být privilegovaný uživatel
 - knihovna „raw sockets“, libnet
 - přímo pro tento účel vytvořený nástroj scapy
 - síťové analyzátory – umí vytvářet předem definované pakety
- hlavní ochrana – sequence number

Morrisův útok

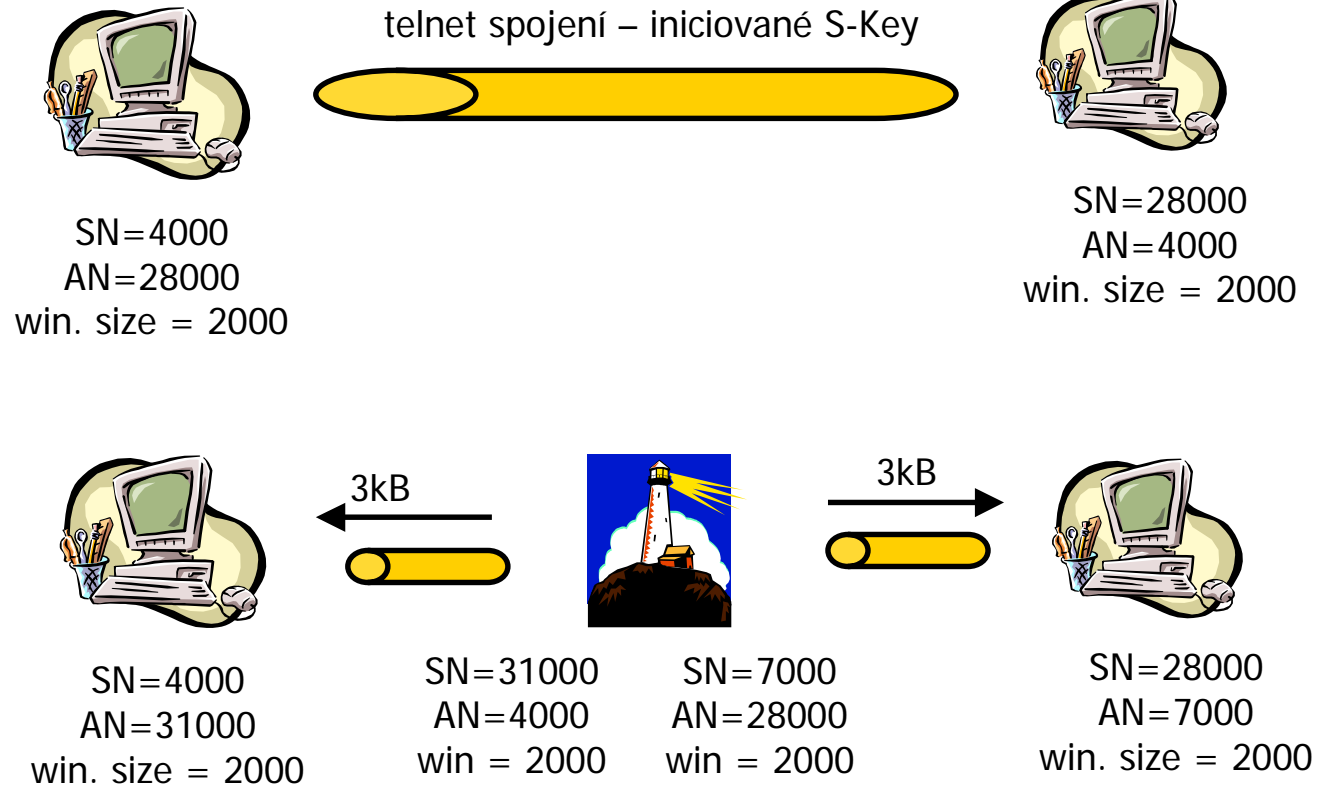
- útočník pošle SYN paket s IP adresou důvěryhodného klienta
- uhodne SN, které server použije v odpovědi – paket samotný útočník nevidí
- SN použije pro dokončení spojení ACK

- nyní může na server posílat příkazy, za jejichž odesilatele je považován důvěryhodný klient
- je třeba
 - zjistit sekvenční číslo
 - eliminovat důvěryhodného klienta – SYN flood

Aktivní útok

- umožňuje přesměrování spojení
- ruší bezpečnost S-key, Kerberosu, ...
- předpoklady
 - útočník vidí pakety mezi klientem a serverem – stačí být na stejném segmentu (před switchem), nebo na stejné bezdrátové síti
- postup
 - oboustranně se desynchronizuje se spojení
 - vytvoření nových dvou spojení s útočníkem uprostřed

Aktivní útok



Slipping in the Window

- Paul A. Watson: TCP Reset Attacks, 2003
- čas potřebný k vložení platného paketu
 - min. velikost paketů (40B), window=64kB
 - 56 kbps 374 s
 - 256 kbps (DSL) 81 s
 - 11 Mbps (WiFi) 2,2 s
 - 100 Mbps (Eth LAN) 0,15 s
 - uhodnutí obou sekvenčních čísel
 - 11 Mbps (WiFi) $1,3 \cdot 10^5$ s
 - 1 Gbps (Eth LAN) $1,3 \cdot 10^3$ s => >20 min
 - window size 1GB
 - 256 kbps (DSL) 0,3 s

SYN varianta

- na paket SYN (na již otevřený port) odpovídá klient obvykle RST
- někdy ale posílá ACK
 - jedna ze stran spojení může sloužit jako reflektor pro DoS útok

Vkládání dat

- nejsložitější možnost využití principů resetovacího útoku
- vložený paket s platnými SN a AN je přijat a nahradí tak některý z originálních paketů
- několik možných efektů
 - změna dat se projeví po ukončení spojení
 - spojení je okamžitě ukončeno (detekce změn)
 - přijímací strana požádá o opakování paketu SSH, SSL/TLS

Implementace

- vlastní testovací aplikace (ročníkový projekt)
- pomocí knihovny libnet
- praktické ověření předchozích tvrzení (RST, SYN)
- omezení rychlosti vytváření paketů na zhruba 15.000 za sekundu
- nepodařilo se přerušit spojení mimo lokální síť (filtrování na bráně vnitřní sítě)
- https – pro každou stránku je otevřeno nové spojení (nový port) => protokol nespolehá na nepřerušitelnost šifrovaného spojení
- vliv firewallu (Win XP) ...

Použití SCAPY

```
>>> ip = IP()
>>> tcp = TCP()
>>> ip <IP |>
>>> tcp <TCP |>
>>> ip.dst = "192.168.9.1"
```

```
def mkpacket():
... global ack
... ack = ack + 1
... p = ip/TCP(ack=ack, flags=r)
... return p
```

```
while ack<CONST:
... res = sr1(mkpacket())
```

Obrana

- kontrola AN – v testovaných OS nebyla implementována
- ingress filtering – kontrola odchozích paketů (možný požadavek na ISP)
- výběr počátečních sekvenčních čísel
- kryptografické zabezpečení
- IPSec
- detekce útoků IDS

Závěr

- původní cíl útoků byl protokol BGP (páteří protokol mezi bránami)
- je možné ho použít v kombinaci s dalšími technikami pro menší útoky (DNS cache poisoning)
- nové úpravy TCP protokolu (IETF drafts)
 - Defending TCP against Spoofing Attacks
 - Improving TCP Robustness to Blind In-Window Attacks

Neexistuje univerzální obrana!



Questions & Comments
