Serwer SSH

Serwer SSH

- Wprowadzenie do serwera SSH
- Instalacja i konfiguracja
- Zarządzanie kluczami

Serwer SSH - Wprowadzenie do serwera SSH

- Praca na odległość – potrzeby w zakresie bezpieczeństwa
  - Identyfikacja i uwierzytelnienie osoby
  - Uwierzytelnienie serwera
  - Zabezpieczenie przed podszyciem się w trakcie trwania sesji
  - Poufność danych przekazywanych podczas pracy (sesja szyfrowana)

Serwer SSH - Wprowadzenie do serwera SSH

- Ogólne cechy SSH
  - Protokół SSH działa w warstwie sesji (5) modelu ISO/OSI, a w warstwie aplikacji modelu TCP/IP.
  - Opiera się na protokole transportu strumieniowego (TCP).
  - Dodatkowo zapewnia kompresje.
  - Zapewnia tunelowanie sesji terminala.
  - Zapewnia tunelowanie protokołu X11.
  - Zapewnia tunelowanie innych protokołów.
  - Zapewnia negocjacje algorytmów kryptograficznych

Serwer SSH - Wprowadzenie do serwera SSH

- Zastosowania SSH
  - Zdalna praca na terminalu tekstowym także graficznym dzięki tunelowaniu X11.
  - Dostęp do serwerów pozbawionych monitora i klawiatury.
  - Tunelowanie dowolnych połączeń TCP:
    - zabezpieczenie niezabezpieczonych protokołów (alternatywa dla TLS),
    - pokonanie firewalli, dostęp do maszyn w wewnętrznej sieci.
  - Bezpieczny transfer plików (przy pomocy dodatkowych protokołów).

Serwer SSH - Wprowadzenie do serwera SSH

- Składniki SSH
  - Protokół transportowy
    - Zapewnia uwierzytelnienie serwera, poufność danych i ich integralność, opcjonalnie obsługuje kompresje.
  - Protokół uwierzytelnienia użytkownika
    - Zapewnia uwierzytelnienie klienta.
  - Protokół połączenia
    - Multipleksuje różne połączenia

Serwer SSH - Wprowadzenie do serwera SSH

- Uwierzytelnianie
  - none – pusta metoda autentyfikacji, każdy jest wpuszczany lub zatrzymywany, zazwyczaj zablokowana,
  - publickey - (klient wysyła sygnaturę zaszyfrowaną swoim kluczem prywatnym, serwer ja odszyfrowuje odpowiednim kluczem publicznym i wpuszcza, jeśli odszyfrowanie się powiodło),
  - password - (klient przesyła do serwera zaszyfrowane hasło),
  - hostbased - (klient jest wpuszczany, gdy określony użytkownik loguje się z określonego komputera /pliki .shosts w katalogu domowym użytkownika po stronie serwera/).

Serwer SSH - Wprowadzenie do serwera SSH

- Osoba używająca elektronicznego podsłuchu zobaczy zaszyfrowane informacje, których odczytanie jest możliwe tylko wtedy, gdy się posiada klucz kryptograficzny.
- Protokół ssh opiera się na kryptografii klucza publicznego.

Klucze: publiczny oraz prywatny

- Klucz publiczny jest powszechnie dostępny (może być umieszczony na stronie sieci Web),
- Klucz prywatny musi być dobrze chroniony.
- Każda kombinacja klucz prywatny/klucz publiczny jest niepowtarzalna.
- Klucz prywatny nie jest przesyłany przez sieć.
- Gdy dane są zaszyfrowane za pomocą klucza publicznego, odszyfrować je można tylko za pomocą klucza prywatnego tego użytkownika (ssh regularnie zmienia swój prywatny klucz, tak aby dane były inaczej szyfrowane co kilka minut).

Serwer SSH - Zarządzanie kluczami

- Uwierzytelnienie serwera – klucze maszyn
  - Każdy serwer posiada klucz maszyny (co najmniej jeden).
  - Dwa modele wiązania kluczy z maszynami
    - Lokalna baza danych u klienta,
    - Zarządca certyfikatów.

Serwer SSH - Instalacja i konfiguracja

- W Linuksie jest używana bezpłatna implementacja o nazwie OpenSSH.
- Protokół ssh pracuje w architekturze klient-serwer.

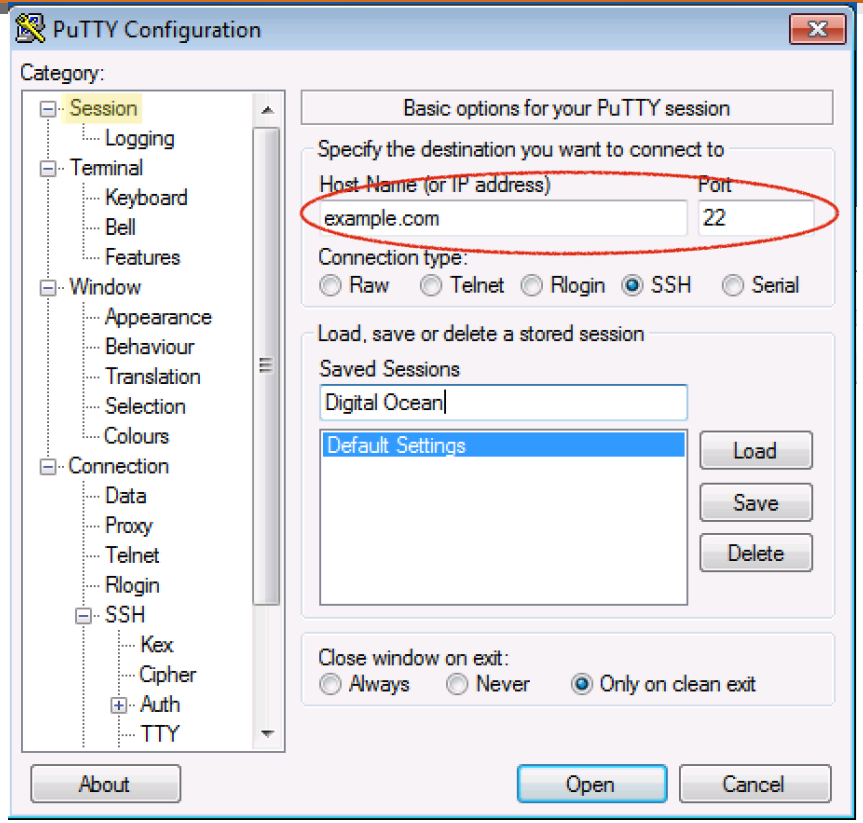
Serwer SSH - Instalacja i konfiguracja

- Instalacja - w dowolnym momencie
  - apt-get install openssh-server
- Konfiguracja
  - /etc/ssh/sshd_config
- Opcje

  # What ports, IPs and protocols we listen for
  - Port 2122

  # Authentication:
  - LoginGraceTime 120
  - PermitRootLogin no
  - AllowUsers user1 user2

Serwer SSH - Instalacja i konfiguracja

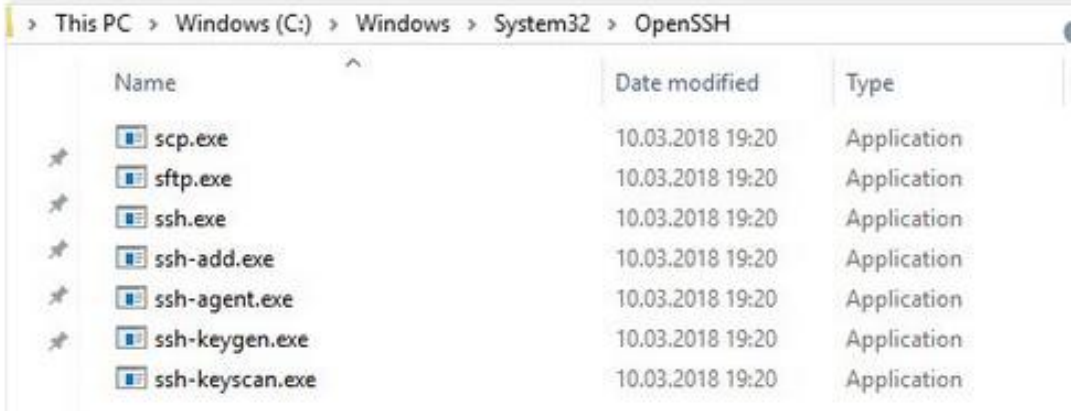


Środowisko Windows

- Nie ma klienta ssh do wersji systemu Windows 10 - jeżeli użytkownik zamierza korzystać w tego protokołu, to musi z sieci pobrać klienta.
- W Windows 10 w wersji 1803, klient OpenSSH, bazujący na oficjalnej wersji 7.6p1 OpenSSH, został zaimplementowany jako funkcja Windows, możliwe jest uwierzytelnienie za pomocą pary kluczy SSH bez Putty i innych programów.
  Microsoft się tym nigdzie głośno nie chwalił – *odkrycia tego dokonał ekspert od bezpieczeństwa Oddvar Moe z firmy Advania Norway*.

Środowisko Windows

- pliki wykonywalne są przechowywane w katalogu C:\Windows\System32\OpenSSH. Wśród nich znajdziemy narzędzia scp i sftp, a także wszystko co potrzeba do zarządzania kluczami. Po pierwszym połączeniu ze zdalnym serwerem, klient zapisze jego odcisk w pliku %UserProfile%\.ssh\known_hosts.



Serwer ssh uruchamiany z system

- Jeżeli serwer ssh ma być uruchamiany podczas startu systemu, to trzeba dołączyć go do listy serwisów uruchamianych automatycznie.

Nawiązania połączenia z serwerem

- W celu nawiązania połączenia z serwerem za pomocą ssh należy wpisać polecenie
  ssh konto@nazwa_serwera lub ssh konto@adres ip.
- Przykładowo polecenie ssh root@192.168.0.224 spowoduje nawiązanie połączenia za pomocą protokołu ssh z komputerem o adresie 192.168.0.224 i zalogowanie użytkownika root (po poprawnym wpisaniu hasła).

- Podczas pierwszego nawiązania połączenia między komputerami jest generowana para kluczy zgodnie z algorytmem szyfrowania RSA.

Zastosowania SSH – transfer plików

- Zwykły FTP tunelowany przez SSH.
- SCP (secure copy) – tylko kopiowanie plików. Zależny od systemu (np. dopasowywanie nazw plików robi system operacyjny serwera).
- SFTP (SSH file transfer protocol) – posiada rozbudowaną funkcjonalność (listowanie katalogów, usuwanie plików, zmiana atrybutów) pozwalającą nawet na zamontowanie zdalnego systemu plików (SSHFS). Mniej zależny od systemu.
- rsync – protokół synchronizacji plików (katalogów), jako warstwy bezpieczeństwa można użyć SSH.

Zastosowania SSH

– tunelowanie –L

ssh -N -L 8888:www.net.pl:80 username@remotehost

Otwiera port 8888 na lokalnym komputerze, połączenie z którym tak naprawdę łączy z www.net.pl:80

– tunelowanie -R

ssh -N -R 8888:serwis.w.mojej.sieci:80 username@remotehost

Otwiera port 8888 na remotehost, połączenie z którym tak naprawdę łączy z serwis.w.mojej.sieci:80

Możliwości konfiguracji serwera OpenSSH

- Wybór dostępnych metod uwierzytelnienia
- Wybór dostępnych algorytmów kryptograficznych i kompresji
- Dostępność lub nie tunelowania X11
- Blokowanie tunelowania określonego typu (domyślnie -R zablokowane dla interfejsów sieciowych innych niż loopback)
- ...
- Sekcje reguł dla poszczególnych użytkowników i hostów, z których się łączą.

Koniec

- Dziękuję za uwagę