Bilibioteki dynamicznie ładowane i współdzielone (Dynamically Linked Libraries – DLL, Shared Objects - SO)

Lazy/Late Binding (Loading, Linking…) itp.

- podczas uruchamiania ładowany do pamięci jest tylko program główny oraz tablice dołączone do w plików wykonywalnych ELF (Linux):

  - *Procedure Linkage Table* (PLT),
  - *Global Offset Table* (GOT), ew. procedura linkera dynamicznego,

- kod danej funkcji bibliotecznej ładowany jest do pamięci **dopiero podczas pierwszego wywołania** tej funkcji w programie.

Zasada działania:

wszystkie wywołania funkcji bibliotecznych linkowanych dynamicznie odbywają się pośrednio: poprzez tablicę PLT oraz tablicę GOT – zawierającą adresy funkcji.

Pierwsze wywołanie funkcji:

1) Instrukcja CALL nie wywołuje bezpośrednio funkcji, ale prostą procedurę (*stub*) w tablicy PLT. Rozkaz CALL zachowuje na stosie adres powrotu do programu wywołującego daną funkcję.

2) Wykonywany jest skok pod adres odczytany z odpowiedniego miejsca tablicy GOT.

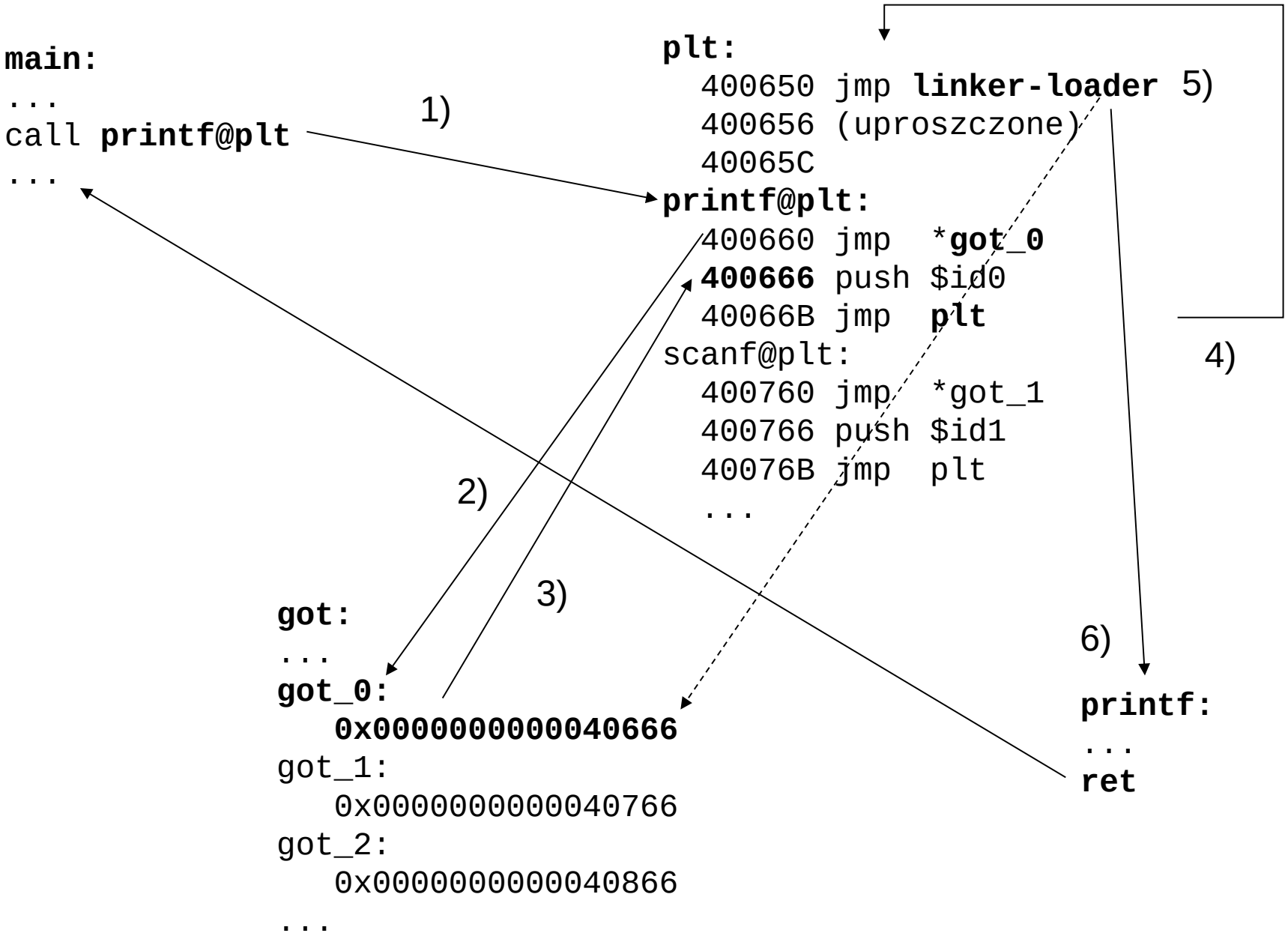
3) Przy pierwszym wywołaniu danej funkcji adresem tym jest adres kolejnej instrukcji w PLT.

4) Na stosie umieszczany zostaje numer-identyfikator żądanej funkcji bibliotecznej, następnie wywoływany jest linker dynamiczny.

5) Żądana funkcja jest ładowana z pliku do pamięci, a odpowiadający jej adres w GOT zamieniany na właściwy: prowadzący do miejsca, gdzie funkcję załadowano.

6) Funkcja zostaje wykonana, kończy ją rozkaz RET i następuje powrót i kontynuacja programu wywołującego.

Stan początkowy – przed pierwszym wywołaniem funkcji



Zasada działania:

drugie i kolejne wywołania tej samej funkcji:

1) Instrukcja CALL ponownie nie wywołuje bezpośrednio funkcji, ale wykonuje skok do PLT. Rozkaz zachowuje na stosie adres powrotu do programu wywołującego funkcję.

2) W PLT wykonywany jest skok pod adres odczytany z odpowiedniego miejsca tablicy GOT.

3) Przy kolejnym wywołaniu funkcji adresem skoku jest już miejsce poprzednio załadowanej funkcji.

4) Funkcja zostaje wykonana, ponieważ już rezyduje w pamięci, drugie i kolejne wywołania przebiegają szybciej od pierwszego.

Kolejne wywołanie tej samej funkcji bibliotecznej