
S10/L3 Epicode Cybersecurity

Assembly x86

**Nell'esercizio di oggi ci
viene richiesto di analizzare
un codice in Assembly x86
ed identificare le varie
funzioni presenti.**

-
- **0x00001141 <+8>:** `mov EAX,0x20` : imposta il valore esadecimale 0x20 (o 32 in decimale) nel registro EAX;
 - **0x00001148 <+15>:** `mov EDX,0x38` : imposta il valore esadecimale 0x38 (o 56 in decimale) nel registro EDX;
 - **0x00001155 <+28>:** `add EAX,EDX` : aggiunge il contenuto del registro EDX al registro EAX e memorizza il risultato in EAX. Quindi, EAX conterrà il valore $32 + 56 = 88$;
 - **0x00001157 <+30>:** `mov EBP, EAX` : copia il valore di EAX in EBP;
 - **0x0000115a <+33>:** `cmp EBP,0xa` : compara il contenuto del registro EBP con il valore esadecimale 0xA (o 10 in decimale);
 - **0x0000115e <+37>:** `jge 0x1176 <main+61>` : salta all'indirizzo 0x1176 (etichettato come <main+61>) visto che il confronto precedente ha dato esito positivo, ovvero se il contenuto di EBP è maggiore o uguale a 10; se il confronto fosse avesse dato esito positivo, il programma avrebbe eseguito le azioni seguenti;
 - **0x0000116a <+49>:** `mov eax,0x0` : imposta il valore 0x0 (o 0 in decimale) nel registro EAX;
 - **0x0000116f <+54>:** `call 0x1030 <printf@plt>` : Chiama la funzione di libreria printf per stampare il contenuto del registro EAX, che è 0.
-