Relatório sobre a análise do código dado.

TAG de engenharia reversa

UFRJ

Thais Angelo Ferreira de Oliveira

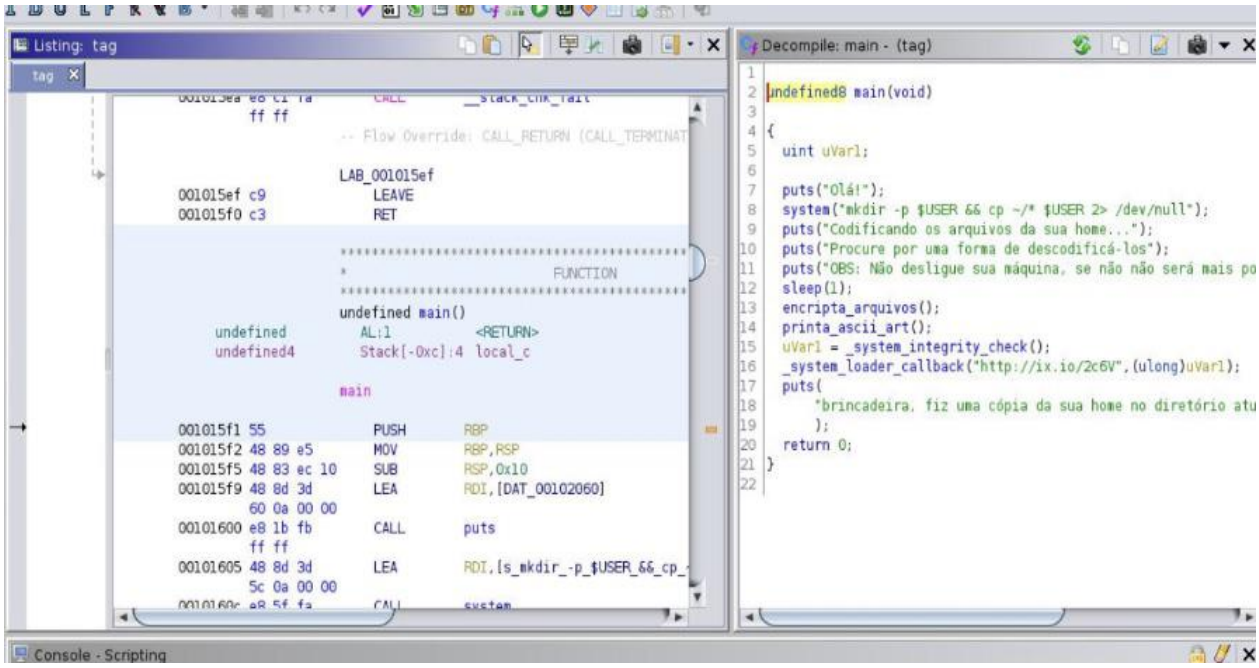
Código foi analisado utilizando-se o software Ghidra sendo possível ver o código Assembly e também o mesmo decompilado na linguagem C.

O arquivo enviado é de extensão ELF, ou seja, por padrão é um arquivo executável.

De início, foi feita uma análise estática do código, estudando as funções existentes nele, o que elas fazem e retornam, onde elas são chamadas, quais strings foram criadas e aonde estão sendo utilizadas. Para esse estudo foram utilizadas algumas ferramentas existentes no ghidra, como a Symbol Table, que mostra todas as funções do programa, a defined string, as referências de cada string entre outras.

Com esse estudo, foi possível observar a funcionalidade do programa, ao executa-lo ele cria um novo diretório no computador com o nome da pasta home, criptografa todos os arquivos contidos nessa pasta e exclui os originais.

Ao explorar o código, foi verificado a existência de uma função main.



Ao analisa-la, é possível verificar que ela é responsável por criar um diretório com o nome da pasta home. Nela também é chamada as funções encripta_arquivos() , printa_ascii_art(), system_integrity_check() e system_loader_callback(), sendo elas as principais do programa.

Começando a analise pela função “system_loader_callback()”, sendo a principal dentro da main e do programa, sua funcionalidade é baixar o arquivo com a encriptação (essa ação é feita pela função “download_file_from_url”) localizado na URL passada como parâmetro e o executa de acordo com o segundo parâmetro passado. A segunda função chamada dentro dela é a “sprintf()” que é responsável por encriptar cada diretório com o arquivo baixado, somando os caracteres de cada arquivo com a chave passada como parâmetro.

A outra função existente na main é a “_system_integrity_check()” é responsável por fornecer a chave que será passada como parâmetro para encriptar os arquivos existentes na home.

No programa também foi visto uma sequência de chamadas de funções em que seus nomes começavam do mesmo modo “curl_easy”. Algumas dessas funções:

- “curl_easy_getinfo”: Responsável por buscar informações do curl.

- “curl_easy_setopt”: Responsável por estabelecer os padrões de retorno e transferência de dados. No caso do programa o endereço passado é http://ix.xo/2c6v .
- “Curl_easy_init”: Retorna um identificador curl.