Contenido

1. Conceptos Básicos Y Expectativas Del Curso. 7
1.1 Funcionamiento De Windows, Mensajes Y Eventos. 10
1.2 ¿Para qué podemos usar el Lenguaje Ensamblador?. 11
1.3 Numeración y Cálculo Aritmético. 11
1.3.1 Números Hexadecimales. 12
1.3.2 Conversiones Decimal – Hexadecimal. 16
1.3.3 Números Negativos. 16
1.3.4 Bits, Bytes, Palabras y sistema Binario. 17
1.3.5 Registros como variables. 18
1.4 RESUMEN AUTOEVALUACION. 24
1.5 Ejercicios. 24
1.5.1 Resultados. 25
2. Lenguaje Ensamblador. 27
2.1 Ejercicio. 33
2.1.1 Resultados. 34
2.2 Descarga E Instalación de MASM32 / Easy Code. 36
2.2.1 Descarga Masm32. 36
2.2.2 Descarga Easy Code. 40
2.2.3 Configuración Easy Code. 43
2.3 Registros Del Sistema 32Bits. 45
2.4 Directivas Del Lenguaje, Estructura Del Programa. 47
2.4.1 Ejercicio Guiado Hola Mundo. De Debug A Microsoft Windows 32bits. 49
2.4.2 Ejercicio: 57
2.5 Tipos De Datos. 58
2.5.1 DB. 58
2.5.2 DW/Word 59
2.5.4 DD/DWord. 60
2.5.5 DQ/QWord 61
2.5.6 DT 62
2.6 Introducción A Las API's De Windows. 63

2.6.1 Donde buscar información sobre API..... 65
2.6.2 Como Agregar API (DLL) A Su Proyecto..... 66
2.7 Mover Datos A Registros Y Viceversa..... 67
2.7.1 Instrucción Mov..... 67
2.8 Operaciones Matemáticas Simples..... 70
2.9 Operaciones De Pila..... 71
3. Lenguaje Ensamblador: Procedimientos, Definición y Uso..... 72
4. Lenguaje Ensamblador: Operadores Y Directivas Relacionadas Con Los Datos..... 75
4.1 Offset..... 75
4.2 Addr..... 75
4.3 PTR..... 75
4.4 Type..... 76
4.5 SizeOf..... 77
5. Lenguaje Ensamblador: Operaciones Con Banderas..... 78
6. Lenguaje Ensamblador: Instrucciones De Desplazamiento..... 79
6.3 Multiplicar Por Desplazamiento..... 81
6.3.2 SHL, Desplazamiento Lógico A La Izquierda..... 82
6.3.3 SHR, desplazamiento lógico a la derecha..... 85
7. Lenguaje Ensamblador: Instrucciones De Transferencia De Control..... 86
7.3 Incondicionales..... 86
7.3.2 JMP..... 86
7.3.3 Invoke..... 86
7.3.4 RET..... 86
7.4 Condicionales Tradicionales..... 86
7.5 Condicionales MASM32..... 88
7.6 Iterativas Tradicionales..... 89
7.7 Iterativa MASM32..... 89
8. Lenguaje Ensamblador: Instrucciones Manejo De Cadenas..... 90
8.3 Prefijos De Repetición..... 90
8.4 Mover Cadenas..... 90
8.4.2 LEA, Cargar dirección Efectiva..... 92
8.5 Comparar Cadenas..... 93
8.6 Buscar En Cadenas..... 94
8.7 Transferencias entre cadenas y registros..... 96

8.7.2 Incrementar Y Decrementar en Uno. 98
9. Modos De Direccionamiento 101
10. Resumen Y Fase De Video Talleres 102
10.3 Ejercicios Varios Para MASM32 103
10.3.2 Resultados. 104
10.4 Proyecto Final MASM32. 108
Primer Objetivo. 108
Segundo Objetivo. 108
Objetivo Tres. 109
10.4.2 Resultado 110
11. Anexo I – Integración Con Leguajes De Alto Nivel. 111
11.1 Como realizar DLL en ensamblador. 111
11.1.1 Creación De DLL En Ensamblador. 112
11.1.2 Creación De DLL En Ensamblador Función para VB.NET. 115
11.1.3 Creación De DLL En Ensamblador Función para Python. 116
11.1.4 Creación De DLL En Ensamblador Funciones A Exportar. 116
11.2 Python Integración. 118
11.3 La Comunidad De Python. 118
11.3.1 Creando Su Propia Biblioteca – Shell Inversa Para Windows Desde Python. 119
11.4 VB.NET Integración 122

1.0 Introducción 127
1.1 ¿Qué es el Reversing o Ingeniería Inversa?..... 128
1.2 ¿Qué es un compilador? 128
1.2.1 Código Fuente. 128
1.2.2 Código Intermedio..... 128
1.2.3 Código Objeto..... 128
1.3 Limitaciones. 129
1.4 ¿Qué dice la Ley, respecto al Reversing?. 129
1.5 Ejercicios..... 129
2.0 Introducción a OllyDBG. 131
2.0 Desensamblador/Código. 133
2.1 Registros. 134
2.1.1 Registros Del Procesador. 134
2.1.2 Flag O Banderas..... 134
2.1.3 Registros De Punto Flotante..... 135
2.2 Dump. 135
2.3 Pila/Stack. 135
2.4 Relación De Teclas Y Botones Más Usados. 136
3.0 Reconstrucción De Código Nativo. 137
3.1 Código Nativo, Variables y Estructuras. 138
3.1.1 Variables. 138
3.2 Ejecutando Código Nativo, Con OllyDBG. 142
3.2.1 Ejecución Completa. 142
3.2.2 Ejecución Línea a Línea..... 143
3.2.3 Pasar Por Encima, Ejecutar Funciones Sin Entrar Dentro De Ellas. 145
3.3 Código Nativo, Variables y Estructuras II. 145
3.3.1 Variables II, sumando. 145
3.3.2 Puntos De Ruptura. 146
3.3.3 Estructuras. 148
3.3.4 Buscando En La Memoria, Sección Del Dump. 149
3.4 Procedimientos Y Variables Locales. 151
3.5 Estructuras De Control Condicionales. 154
3.5.1 Instrucciones De Transferencia De Control Según los Flag. 158

3.6 Estructuras De Control Iterativas. 159
3.7 Estructuras De Control Repetitivas. 162
3.8 Formularios. 166
3.9 Ficheros Binarios PE. 178
3.9.1 Diseño. 178
3.9.2 Tabla De Secciones. 179
3.9.3 Tabla IMPORT. 180
3.9.4 Relocalizaciones. 180
4.0 API'S Significativas. 182
5.0 Practicas Con Supuestos En Código Nativo. 193
5.1 Buscando Cadena En Código Nativo. 193
5.2 Poniendo Parches ("Patches"). 202
5.3 Ejercicio. 204
5.4 Cifrado De Texto Por XOR. 205
5.4.1 Cifrado Mediante XOR. 207
5.5 Anular Objetivo Por API (Intermodular Calls). 208
5.6 Ejercicios. 211
5.7 Analizando A Shellter, Un Malware Real. 212
5.7.1 Analizando La Calculadora De Windows Sin Infectar, Mapa Memoria. 213
5.7.2 Analizando La Calculadora De Windows Infectada, Mapa Memoria. 216
5.7.3 Analizando La Calculadora De Windows Sin Infectar, Hilos. 220
5.7.4 Analizando La Calculadora De Windows Infectada, Hilos. 221
5.7.5 Analizando La Calculadora De Windows Infectada, Hilo Principal Shellter. 225
6.0 Reconstrucción De Código Intermedio. 229
6.1 MicroSoft Intermediate Language 229
6.1 Analizando Binario De Código Intermedio VB.NET. 230
6.1.1 Analizando Binarios VB.NET, con OllyDBG. 230
6.1.2 Ejercicio. 236
6.3 ExeInfoPe. 236
6.4 Introducción A .Net Reflector. 238
7.0 Practica Con Supuesto En Código Intermedio. 239